



FACULTAD DE INGENIERÍAS

GRUPO DE INVESTIGACIÓN EN ASTRONOMÍA ALFA ORIÓN

Evaluación de las Capacidades Científicas y Didácticas del Radiotelescopio en Banda KU del Observatorio Astronómico UTP

Trabajo de grado presentado por:

David Galeano Gómez
Miguel Ángel Salazar Manzano

COMO REQUISITO PARCIAL PARA OPTAR POR EL TÍTULO DE INGENIERO
FÍSICO

Director del Trabajo de grado:
MSc. Edwin Andrés Quintero Salazar

Evaluación de las Capacidades Científicas y Didácticas del Radiotelescopio en Banda KU del Observatorio Astronómico UTP

DAVID GALEANO GÓMEZ
MIGUEL ÁNGEL SALAZAR MANZANO

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
Facultad de Ingenierías
Programa de Ingeniería Física

Pereira, Noviembre de 2020

Dedicatoria

Dedicamos este trabajo a nuestra familia. Especialmente a nuestros padres, quienes con su apoyo y esfuerzo nos llevaron adelante creyendo siempre en nuestra capacidad.

Agradecimientos

Queremos agradecer a nuestro director de tesis MSc. Edwin Andrés Quintero Salazar, quien fue un guía esencial durante todo el proceso de desarrollo de este trabajo. Su confianza y apoyo hicieron posible la realización de tan ardua labor. Igualmente queremos agradecer al grupo de Investigación en Astronomía Alfa Orión, cuyos miembros fueron un pilar en la exploración de las capacidades didácticas del instrumento.

Finalmente queremos agradecer a nuestra familia y amigos por estar presentes durante este largo camino, impulsando nuestro ánimo cuando las energías se vieron agotadas. Gracias, totales.

Contenido

Índice de figuras	7
Resumen	11
1 Introducción	12
1.1 Problema	12
1.2 Justificación	13
1.3 Objetivos	14
1.3.1 Objetivo General	14
1.3.2 Objetivos Específicos	14
1.4 Estado del Arte	14
2 Radiotelescopio en banda KU del OAUTP	22
2.1 Descripción	22
2.2 Puesta a punto	26
2.2.1 Cambio de la caja de distribución	26
2.2.2 Actualización del firmware del controlador y modificación en los parámetros para la operación	27
2.2.3 Unificación de los circuitos	28
3 Calibración	30
3.1 Calibración del apuntado	30
3.1.1 Error instrumental	30
3.1.2 Cuantificación del error instrumental en acimut y altitud	31
3.1.3 Ventanas de observación	32
3.1.4 Calibración en el apuntado del radiotelescopio del OAUTP	33
Primera etapa (Ajuste grueso)	34
Segunda etapa (Ajuste fino)	36
3.2 Calibración del plato reflector	40
3.2.1 Temperatura de operación	40

<i>CONTENIDO</i>	5
3.2.2 Ancho de Haz	42
3.2.3 Area efectiva, Eficiencia y Ganancia	45
3.2.4 Sensibilidad aproximada	46
4 Software	49
4.1 Interfaz gráfica de usuario	50
4.1.1 Ventana Adquisición de datos	50
4.1.2 Ventana Escaneo	51
4.1.3 Ventana Calibración	53
4.2 Algoritmos	54
4.2.1 Ventana Adquisición de datos	54
4.2.2 Ventana Escaneo	58
4.2.3 Ventana Calibración	63
4.2.4 Comunicación DAQ	67
4.2.5 Envío y recepción de coordenadas	69
4.2.6 Ajuste Gaussiano	73
5 Potencial científico del instrumento	74
5.1 Ventanas de observación	75
5.1.1 Técnica matricial	75
5.1.2 Técnica de escaneo	76
5.2 Tránsitos	77
5.3 Temperatura de brillo	77
6 Estrategia didáctica para la enseñanza de la radioastronomía	80
6.1 Parámetro físico a observar	80
6.2 Estrategia didáctica socio constructivista	81
6.3 Técnicas de recolección de la información	82
6.3.1 Observación participante	82
6.3.2 Grupo de discusión	82
6.4 Instrumento para la enseñanza de la radioastronomía	82
6.4.1 Desarrollo de la guía	83
Cálculo del HPBW	86
Cálculo del cociente entre el área angular del haz de la Antena y el área angular del objeto observado	86
Cálculo de la temperatura de brillo del Sol	87

7	Resultados	88
7.1	Capacidades científicas del radiotelescopio	88
7.1.1	Observación de radiofuentes de muestra	88
	Sol	88
	Luna	90
	Centro galáctico	93
	Saturno	94
	Cassiopeia A	95
	Cygnus A	96
	Taurus A	97
	Nebulosa de Orión	98
7.1.2	Flujo y temperatura de brillo	99
7.2	Capacidades didácticas del radiotelescopio	102
8	Conclusiones	106
A	Estrategia didáctica	111
A.1	Observación participante	111
A.2	Grupo de discusión	113
A.3	Guía aplicativa	115
B	Interfaz Gráfica de Usuario	127
B.1	Guía de usuario	127
B.2	Código fuente funciones estructurales “CRT.py”	171
B.3	Código fuente entorno gráfico “Adquisicion.py”	201
B.4	Código fuente widget-matplotlib “mplwidget.py”	215

Índice de figuras

2.1 Radiotelescopio OAUTP.	22
2.2 Diagrama del radiotelescopio OAUTP.	23
2.3 Radiorreceptor.	23
2.4 Sistema de movimiento.	24
2.5 Software de control de la montura.	24
2.6 Resultados arrojados por el radiotelescopio.	25
2.7 Tránsito solar bajo condiciones climáticas diferentes.	25
2.8 Caja de distribución.	27
2.9 Esquema para el ajuste de la potencia de los motores.	28
2.10 Conexiones para la adquisición de datos.	29
2.11 Diagrama circuital de la caja de alimentación y conexiones.	29
3.1 Representación gráfica del error en el apuntado.	31
3.2 Ventana de observación resultante dentro del apartado de calibración.	33
3.3 Representación gráfica de las etapas de ajuste grueso y ajuste fino.	34
3.4 Ventana de calibración para el Sol con un tamaño de $12^\circ \times 12^\circ$ y una resolución de 1° . En ella se aprecia el Sol desviado del centro de la imagen.	35
3.5 Ventana de calibración para el Sol con un tamaño de $12^\circ \times 12^\circ$ y una resolución de 1° . En ella el Sol se encuentra centrado en la imagen	36
3.6 Ventanas de calibración para el Sol con un tamaño de $2^\circ \times 2^\circ$ y una resolución de 0.2°	37
3.7 Desviación encontrada para la ventana actual y promedio de todas las desviaciones.	38
3.8 Ventana de calibración para el Sol con un tamaño de $2^\circ \times 2^\circ$ y una resolución de 0.2° , en ella se observa el correcto apuntado de la antena.	38
3.9 Error en el apuntado al realizar cambio de hemisferio.	39
3.10 Patrón de radiación de la antena y ancho de haz.	43
3.11 Tránsito solar y ajuste Gaussiano.	43
3.12 Respuesta del radiotelescopio ante diferentes radiofuentes.	48

4.1	Estructura <i>Compact Radio Telescope UTP</i>	49
4.2	Página: Adquisición de datos.	50
4.3	Página 2: Escaneo.	52
4.4	Página: Calibración.	53
4.5	Diagrama de flujo al oprimir el botón “Cielo de fondo”.	55
4.6	Diagrama de flujo al oprimir el botón “Referencia”.	56
4.7	Diagrama de flujo al oprimir el botón “Tránsito”.	57
4.8	Diagrama de flujo al oprimir el botón “Ver”.	58
4.9	Ventana Escaneo: Primera parte.	59
4.10	Ventana Escaneo: Segunda parte.	60
4.11	Ventana Escaneo: Tercera parte.	62
4.12	Ventana Calibración: Primera parte.	64
4.13	Ventana Calibración: Segunda parte.	65
4.14	Ventana Calibración: Tercera parte.	66
4.15	Ventana Calibración: Cuarta parte.	67
4.16	Diagrama de flujo de la función <i>ADQ()</i>	68
4.17	Ventana principal Pstrotator.	69
4.18	Diagrama de flujo función <i>Position()</i>	70
4.19	Función <i>SendPosition()</i> : Primera parte.	71
4.20	Función <i>SendPosition()</i> : Segunda parte.	72
4.21	Diagrama de flujo <i>Gauss()</i>	73
5.1	Matriz espacial de las intensidades percibidas.	75
5.2	Direcciones del movimiento.	76
5.3	Coordenada constante bajo valores de acimut diferentes.	77
6.1	Ubicación de la capa solar a observar. Tomado de: https://www.nasa.gov/mission_pages/iris/images/index.html	81
6.2	Toma de Cielo de fondo.	84
6.3	Toma de respuesta a una temperatura de referencia.	85
6.4	Toma del tránsito solar.	86
7.1	Ventana de observación de 12 ° x 12 ° y pasos de 1 ° alrededor del Sol.	89
7.2	Ventana de observación de 4 ° x 4 ° y pasos de 0.4 ° alrededor del Sol (10/29/2020, 15:03).	89
7.3	Escaneo sobre el sol el 11/05/2020 desde las 15:25, durante 30 minutos.	90
7.4	Detección de tránsito solar durante 15 y 30 minutos respectivamente.	90
7.5	Ventana de observación de 4 ° x 4 ° y pasos de 0.4 ° alrededor de la Luna.	91

7.6	Ventana de observación de $4^\circ \times 4^\circ$ y pasos de 0.2° alrededor de la Luna (10/03/2020, 05:27).	91
7.7	Escaneo sobre la luna el 10/03/2020 desde las 05:15, durante 30 minutos. . .	92
7.8	Detección de medio tránsito lunar durante 15 minutos.	92
7.9	Ventana de observación de $12^\circ \times 12^\circ$ y pasos de 1° alrededor del centro galáctico.	93
7.10	Detección de tránsito galáctico durante 3 horas.	94
7.11	Ventana de observación de $4^\circ \times 4^\circ$ y pasos de 0.4° alrededor de Saturno. . .	94
7.12	Tránsitos de 15 min sobre Saturno.	95
7.13	Ventana de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° alrededor de Cassiopeia A. .	95
7.14	Tránsitos de 20 min sobre Cassiopeia A.	96
7.15	Ventana de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° alrededor de Cygnus A. .	96
7.16	Tránsitos de 20 min sobre Cygnus A.	97
7.17	Ventana de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° alrededor de Taurus A. .	97
7.18	Tránsitos de 20 min sobre Taurus A.	98
7.19	Ventana de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° alrededor de la Nebulosa de Orión.	98
7.20	Tránsitos de 20 min sobre la Nebulosa de Orión.	99
7.21	Punto de máxima y mínima de detección respecto al cielo de fondo	100
7.22	Desarrollo de la práctica.	105
A.1	Patrón de difracción por una rendija circular.	117
A.2	Patrón de radiación típico en una antena.	118
A.3	Radiación de cuerpo negro de objetos con mismo tamaño angular a diferentes temperaturas.	119
A.4	Comparación del ancho angular con el ancho físico.	121
A.5	Antena de plato reflector.	122
A.6	Controlador SPID MD-02.	122
A.7	Resultado de la adquisición de datos.	125
B.1	Estructura general <i>Compact Radio Telescope</i>	133
B.2	Pestaña Adquisición de Datos.	134
B.3	Ingreso de coordenadas.	135
B.4	Mensaje de advertencia para coordenadas erróneas.	135
B.5	Resultado al oprimir “Cielo de fondo”.	136
B.6	Resultado al oprimir “Referencia”.	137
B.7	Mensaje de espera mientras se ejecuta la adquisición.	138
B.8	Resultado final.	138

B.9 Ajuste Gaussiano durante la sesión.	139
B.10 Ventana Escaneo.	140
B.11 Ingreso de los parámetros.	141
B.12 Resultado final.	142
B.13 Ventana Calibración.	143
B.14 Ingreso de parámetros	144
B.15 Resultado final pestaña Calibración	145
B.16 Resultado final pestaña Calibración tras varias sesiones de adquisición. . . .	146
B.17 Ventana “Acerca de”.	147
B.18 Ventana Adquisición de datos al apuntar el sol el 3/11/2020 a las 14:33. . . .	148
B.19 Ventana Adquisición de datos al apuntar el sol el 3/11/2020 a las 15:07. . . .	148
B.20 Detección del sol con la ventana Escaneo.	149
B.21 Ventanas de calibración alrededor del Sol.	149
B.22 Ventanas de calibración alrededor de Intelsat 30.	150
B.23 Diagrama de flujo al oprimir el botón “Cielo de fondo”.	151
B.24 Diagrama de flujo al oprimir el botón “Referencia”.	152
B.25 Diagrama de flujo al oprimir el botón “Tránsito”.	153
B.26 Diagrama de flujo al oprimir el botón “Ver”.	154
B.27 Ventana Escaneo: Primera parte.	155
B.28 Ventana Escaneo: Segunda parte.	156
B.29 Ventana Escaneo: Tercera parte.	157
B.30 Ventana Calibración: Primera parte.	160
B.31 Ventana Calibración: Segunda parte.	161
B.32 Ventana Calibración: Tercera parte.	162
B.33 Ventana Calibración: Cuarta parte.	163
B.34 Diagrama de flujo de la función <i>ADQ()</i>	164
B.35 Ventana principal Pstrotator.	165
B.36 Diagrama de flujo función <i>Position()</i>	166
B.37 Función <i>SendPosition()</i> : Primera parte.	167
B.38 Función <i>SendPosition()</i> : Segunda parte.	168
B.39 Diagrama de flujo <i>Gauss()</i>	169

Resumen

Los estudios en radioastronomía con antenas compactas se han vuelto cada vez más frecuentes tras la llegada de las comunicaciones a través de las radio frecuencias. Particularmente, en la banda KU se han desarrollado gran cantidad de arreglos compactos para la observación solar y lunar presentando resultados positivos. Sin embargo, observaciones como la del tránsito del remanente de supernova Cassiopeia A con una antena de 2.3 m en en el año 2014, en Senigallia, Italia, y la observación exitosa de Saturno y el brazo de la Vía Láctea en 2016, permiten vislumbrar las capacidades de este tipo de tecnología. Estas observaciones abren la posibilidad para el estudio de otras radio fuentes en este rango del espectro electromagnético. En consecuencia, en este proyecto se realizó un estudio de factibilidad que se desarrollará desde el Observatorio Astronómico de la UTP. En este trabajo se desarrolló la calibración y puesta a punto del radiotelescopio en banda KU disponible en las instalaciones del Observatorio Astronómico UTP, en búsqueda de ampliar las capacidades del instrumento para la observación de diferentes radiofuentes que emiten en el rango de los 10 - 12 GHz. Objetos como Cassiopea A, Saturno, el brazo galáctico, la Luna, entre otros, son candidatos para la observación, a los cuales se les realizó un estudio para determinar si es posible su detección a estas frecuencias. Los resultados arrojados evidencian la viabilidad de observación del Sol, la Luna y el brazo galáctico. Además, se desarrolló el Software CRT-UTP que permite la manipulación y la calibración espacial del radiotelescopio. La aplicación también brinda 3 diferentes formas de observación para los objetos de interés. Dada las características del instrumento se planteó una estrategia didáctica y pedagógica para la enseñanza de la radioastronomía, acercando así esta rama de la astronomía a los estudiantes de radiotelescopio. Finalmente se presentan algunas consideraciones que deben ser tenidas en cuenta durante la manipulación del objeto, como también se proponen algunos cambios en los equipos para aumentar las capacidades de observación del instrumento.

Capítulo 1

Introducción

1.1 Problema

La radioastronomía, a diferencia de la astronomía óptica, no requiere de condiciones observacionales excepcionales para obtener datos científicos útiles de los cuerpos celestes en estudio (Castillo, 2010). En Colombia, la volatilidad de las condiciones meteorológicas dificulta los avances de la astronomía en el espectro de luz visible, por lo que las ondas de radio se constituyen como una alternativa para el desarrollo de la astronomía nacional. Específicamente, en la banda KU (12 - 18 GHz), las ondas electromagnéticas penetran los gases de la atmósfera sin verse afectadas, siendo su único obstáculo las masas densas de agua (Raulin y Pacini, 2005). En los años 90, la banda KU empezó a ser utilizada para la comunicación satelital, lo cual llevó al constante desarrollo de los radios de la época. Así mismo, la investigación amateur en radioastronomía encontró en estos nuevos dispositivos la posibilidad de estudio del universo en estas frecuencias (Saje y Vidmar, 2017). Hoy en día, sistemas de observación construidos con tecnología de comunicación satelital han sido implementados en España (Gómez Tornero et al., 2010), Alemania (Bhatia et al., 2006) y Cuba (Figueredo y Hernández, 2009), arrojando resultados satisfactorios al realizar observación solar a 11 GHz con radiotelescopios compactos.

En Pereira, desde el Observatorio Astronómico de la Universidad Tecnológica de Pereira (OAUTP), con el fin de realizar la observación de cuerpos celestes en altas frecuencias, se construyó en 2018 un radiotelescopio enfocado en la banda KU, haciendo uso de elementos propios de la televisión satelital (Higuita et al., 2019). Sin embargo, aunque la observación solar a 11 GHz se realizó con éxito, la posibilidad de apuntar y estudiar a otras radiofuentes requiere del conocimiento de algunos parámetros específicos del radiotelescopio, y exige una adecuada calibración y puesta a punto del instrumento. Este proyecto planteó la realización de un estudio de factibilidad para la observación de objetos como el remanente de supernova

Cassiopeia A (Falcienelli, 2014), Saturno y el brazo galáctico (Mandal et al., 2016) a través de escaneos de deriva. Para esto es necesario llevar a cabo la calibración espacial del apuntado, a partir de la caracterización de la precisión del radiotelescopio (Hiep et al., 2012), y la identificación y cuantificación de los errores de posicionamiento presentes desde la instalación del equipo (Muth et al., 2011).

Los resultados arrojados por este proyecto permitirán la realización de nuevos estudios en radioastronomía de altas frecuencias desde el OAUTP, facilitando la detección de radiofuentes a través del mapeo de zonas específicas del cielo. Además, el estudio de factibilidad propuesto en este proyecto permitirá sugerir mejoras instrumentales para el radiotelescopio, lo cual incrementará el potencial científico del sistema.

1.2 Justificación

La correcta calibración espacial del radiotelescopio en banda KU del OAUTP, permitirá la implementación de escaneos de deriva en regiones específicas del cielo, aumentando la probabilidad de detectar radiofuentes de menor intensidad y facilitando el apuntado del radiotelescopio hacia objetos de mayor envergadura como la Luna y el Sol. Además, los errores de posicionamiento podrán ser monitoreados con mayor frecuencia, permitiendo identificar el momento pertinente de una nueva calibración. Una vez adelantada la calibración y cuantificación de errores, el análisis de factibilidad para la observación de radiofuentes como Saturno, el brazo de la Vía Láctea y el remanente de supernova Cassiopeia A, permitirá establecer el potencial del instrumento para el estudio de fenómenos físicos como la temperatura media de brillo de objetos como el Sol y radiofuentes no térmicas detectables a 11 GHz. En complemento, una estructuración ordenada del procedimiento involucrado en el estudio de este fenómeno físico, podrá desembocar en el desarrollo de guías y prácticas de laboratorio en radioastronomía, promoviendo la enseñanza de esta rama de la astronomía en Colombia. Finalmente, será posible establecer trabajos futuros que permitan modernizar algunas etapas del radiotelescopio, ampliando su potencial científico. La realización de este proyecto contribuye en la necesidad de enseñanza de la radioastronomía en Colombia, dadas las diversas condiciones climatológicas encontradas en el país, presentando el proceso de calibración de radiotelescopios pequeños, de fácil adquisición para instituciones educativas, y su utilidad para la observación de objetos celestes en la banda KU.

1.3 Objetivos

1.3.1 Objetivo General

Establecer la factibilidad del uso del radiotelescopio de banda KU del OAUTP para la observación de radiofuentes distintas al Sol como Saturno, el brazo de la Vía Láctea y el remanente de supernova Cassiopeia A.

1.3.2 Objetivos Específicos

- Desarrollar una metodología para la calibración y puesta a punto del sistema de apuntado del radiotelescopio.
- Cuantificar el error de posicionamiento en las direcciones de Acimut y Altitud del movimiento del radiotelescopio.
- Ampliar el uso del radiotelescopio a la enseñanza de la radioastronomía a través de la construcción de una práctica de laboratorio.

1.4 Estado del Arte

En la actualidad, la astronomía ha logrado adelantar múltiples estudios alrededor de los cuerpos celestes conocidos, siendo el espectro óptico una herramienta fundamental a la hora de interpretar la información proveniente de la luz emitida por estos. De forma paralela, una de las ventanas de observación más importante, y utilizada hoy en día, se encuentra en la región de radio del espectro electromagnético, donde se desenvuelve la radioastronomía. La radioastronomía tiene sus origen en 1932, siendo su pionero Karl Jansky, el cual descubre un ruido constante de naturaleza desconocida cuando culmina sus estudios enfocados en la detección de la interferencia presente en las comunicaciones radiotelefónicas (Montoya y del Mar, 2011). Aunque este hallazgo pasó inadvertido frente a los astrónomos de la época, en 1937 Grote Reber construye una antena de plato parabólico basado en los diseños de Jansky, logrando detectar ruido cósmico a 160 MHz, lo que le permitió publicar el primer mapa celeste en frecuencias de radio en 1944 en la revista *The Astrophysical Journal*. Los resultados obtenidos con la antena parabólica de 9 m de diámetro implementada por Reber popularizaron el diseño de plato parabólico, lo que impulsó la construcción de radiotelescopios de gran envergadura como el Arecibo¹ de 305 m en 1960, Parkes² de 64 m en 1961 y el Green Bank³ de 43 m en 1967. La construcción de antenas de grandes dimensiones implica múltiples

¹<https://www.naic.edu/ao/>

²<https://www.parkes.atnf.csiro.au/>

³<https://greenbankobservatory.org/>

retos de ingeniería, por lo que en la actualidad la optimización de un radiotelescopio se basa principalmente en técnicas interferométricas (Mendoza Torres, 2003). Esta estrategia implica el uso de múltiples antenas como el arreglo implementado por el Very Large Array⁴ (VLA) en Socorro, Nuevo México, el cual está compuesto por 27 antenas independientes, cada una de las cuales tiene un diámetro de 25 metros. Durante el último siglo la radioastronomía ha hecho factible la realización de nuevos estudios sobre los cuerpos celestes, además de permitir la identificación de otros nuevos como los púlsares, quásares y galaxias activas. Estos aportes no hubieran sido posibles sin la construcción de voluminosas estructuras como las mencionadas anteriormente, por lo que, en un principio, esta rama de la astronomía no se practicaba frecuentemente de forma amateur (Saje y Vidmar, 2017).

En los años 80, los avances en distintos sistemas de comunicaciones, como la telefonía y la televisión, hizo imperante el avance de nuevos medios para transmitir la información. Una solución a esto se encontró al incursionar en la comunicación satelital, logrando transmitir información al rededor del mundo con mayor eficiencia, siendo las antenas el instrumento fundamental para generar este flujo de datos. Es así como en 1983, la cadena televisiva NBC utiliza la banda KU (12 - 18 GHz) para transmitir su contenido, impulsando el desarrollo de radios con un diseño enfocado a esta banda de frecuencias. La tecnología de fácil acceso suministrada por la televisión satelital dio a la radioastronomía amateur un estímulo para adelantar estudios orientados a las altas frecuencias, desde la banda C hasta la banda KU (Bhatia et al., 2006). En estas longitudes de onda centimétricas es posible detectar eventos solares (Figueredo y Hernández, 2009) además de determinar parámetros físicos como la temperatura de brillo de radiofuentes como el Sol (Manisha, 2016) y Cassiopeia A (Falcienelli, 2014) en base a la teoría de radiación de cuerpo negro y la aproximación de Rayleigh-Jeans.

En el año 2006, el Instituto de Fraunhofer para circuitos integrados, en colaboración con la Agencia Espacial Europea, desarrollaron un prototipo de radiotelescopio con propósitos educativos, por lo que fue presentado un diseño simple para ser distribuido en instituciones educativas (Bhatia et al., 2006). El equipo cuenta con una antena de disco de 1.2 m de diámetro, alcanzando una ganancia máxima de 41.5 dB a 11 GHz. La señal obtenida es procesada a través del receptor de medición de antena AMA 301, elemento propio de la televisión satelital de fácil adquisición. Por otro lado, la necesidad de mantener un bajo costo tuvo consecuencias en la calidad del sistema de posicionamiento del instrumento, siendo inapropiado para su aplicabilidad en radioastronomía. El prototipo arrojó resultados positivos al realizar observación solar y lunar a 12.6 GHz. Se logró entonces la detección de un tránsito solar y lunar, además del seguimiento del Sol pese a las limitaciones del movimiento. La ob-

⁴<https://public.nrao.edu/telescopes/vla/>

servación solar permitió establecer las capacidades del sistema y la posibilidad de apuntar otras radiofuentes, identificando los principales limitantes en el sistema de posicionamiento como las altas temperaturas del sistema y la baja resolución del receptor (Bhatia et al., 2006).

Para el 2009, el Instituto de Geofísica y Astronomía de la Habana implementó un prototipo de radiotelescopio centrado en la banda KU con el objetivo de ampliar y dar continuidad a la radioastronomía solar en Cuba (Figueredo y Hernández, 2009). Aunque ya se contaba con dispositivos para observación solar en el rango de las longitudes de onda métricas (235 - 280 MHz) y centimétricas (6700 - 9500 MHz) se buscaba implementar un nuevo dispositivo basado en un sistema LNB de bajo costo y con la mejor relación señal-ruido. Se presentó un radiotelescopio con una antena parabólica de 3 m de diámetro y un sistema de recepción compuesto por un LNB, un buscador de satélites y un amplificador DC, presentando resultados positivos al realizar observación solar a 12.5 GHz (Figueredo y Hernández, 2009). Los autores verificaron el correcto funcionamiento del equipo al comparar registros simultáneos de observación solar a 6.9, 7.5 y 12.5 GHz evidenciando mejoras significativas en la calidad de la señal en esta última frecuencia. La coexistencia de los distintos radiotelescopios ha enriquecido el monitoreo de la actividad solar desde la Habana, permitiendo captar eventos solares a diferentes longitudes de onda desde el observatorio local. En complemento, los investigadores logran el registro de un tránsito solar, lo que permite identificar el ancho del patrón de radiación del sistema colector, para la frecuencia del receptor, y el análisis tanto de la propia actividad solar como de las fluctuaciones del medio interplanetario y atmosférico (Figueredo y Hernández, 2009).

En el 2013, la Universidad de Bolonia, Bolonia, Italia, desarrolló un radiotelescopio compacto centrado en una frecuencia de 12 GHz haciendo uso de elementos propios de la televisión satelital (Tinti, 2013), presentando un sistema de bajo costo con objetivos educativos. El dispositivo desarrollado cuenta con un plato parabólico y un LNB como parte inicial del receptor. Posterior al LNB la señal adquirida es amplificada implementando un sistema de construcción propia y otro de la televisión satelital con el objetivo de alcanzar la mejor calidad de señal posible. Finalmente, esta es filtrada limitando el ancho de banda de la recepción a 200 MHz, logrando así disminuir el ruido de fondo. El dispositivo presentó resultados positivos al realizar observación Solar y Lunar, además de detectar el centro Galáctico desde Italia. Estas capturas se lograron registrando el tránsito de cada una de estas radiofuentes mientras esta cruzaba a través del haz principal de la antena.

Para el año 2016, la Universidad de Pune, India, presentó un radiotelescopio con un plato parabólico de 1.2 m y un LNB, permitiendo captar frecuencias en el rango de 10 y 12 GHz

(Manisha, 2016). El equipo cuenta con un sistema de posicionamiento construido con motores DC y Arduino, facilitando el movimiento en las direcciones de acimut y altitud a través de una interfaz gráfica de usuario desarrollada en MATLAB. Por otro lado, la señal proveniente del LNB pasa por un buscador de satélites, lo que permite visualizar la intensidad de esta, además de facilitar su medición través de un voltímetro. El sistema arrojó resultados satisfactorios, evidenciando el cambio de intensidad en la señal al apuntar el radiotelescopio hacia el Sol directamente. Esta primera lectura motiva la realización del seguimiento, el cual es logrado con éxito evidenciando la aplicabilidad del sistema para el monitoreo de eventos solares (Manisha, 2016). De forma complementaria, el software implementado "Skypipe" permite interpretar la proporcionalidad presente entre la intensidad de respuesta del receptor y temperatura del objeto apuntado, arrojando valores de más de 9000 K al realizar observación solar.

A finales del año 2018, en la Universidad de Hertfordshire, Reino unido, se desarrolló un radiotelescopio en banda KU construido con tecnología de televisión satélital, centrado en una frecuencia de 12.2 GHz para la detección de masas de metanol. La señal obtenida del buscador de satélites es llevada a un espectrometro, con el fin de monitorear las masas de metanol en zonas de alta formación estelar. El sistema logró los resultados esperados al realizar la observación solar, permitiendo caracterizar el ancho de haz de la antena y la temperatura de brillo del sol, arrojando valores de 1° y 11965 K respectivamente (Chanka, 2018). Aunque el proceso de calibración del sistema receptor alcanza resultados satisfactorios, la detección de los cuerpos celestes de interés no es alcanzada. Al comparar los datos obtenidos por el sistema implementado, con las observaciones de masas de metanol realizadas a 6.6 GHz por el Radio telescopio HartRAO de 26 m, se evidencian las limitaciones de investigación por la baja resolución del instrumento. Como consecuencia de esto, la observación de radiofuentes de baja intensidad implica la adquisición de datos durante un tiempo de integración de más de una hora, una magnitud que no puede ser alcanza al implementar tránsitos sobre el objeto. Los autores sugieren realizar, como trabajo futuro, la exploración de las capacidades educativas del instrumento, además de complementar el montaje con una montura motorizada.

En el año 2018, en la Universidad Tecnológica de Pereira, Colombia, se construyó un radiotelescopio con una antena parabólica de 1.2 m, un sistema LNB y una montura altacimutal motorizada con una resolución de 0.2° (Higuita et al., 2019). La señal obtenida desde el LNB es llevada a un buscador de satélites, para posteriormente pasar por un filtro Butterworth pasabajos de segundo orden con frecuencia de corte en 500 MHz, permitiendo así detectar eventos solares. Finalmente se digitaliza la señal a través de una tarjeta de adquisición de datos para facilitar su procesamiento a través de un PC. Se obtuvieron resultados satisfacto-

rios al registrar un tránsito solar bajo diferentes condiciones climáticas, permitiendo validar la utilidad de la antena al trabajar bajo diferentes ambientes. Además, gracias a la resolución del sistema de posicionamiento, se logró realizar seguimiento de forma exitosa, manteniendo el Sol en el haz central de la antena. Sin embargo, los autores sugieren realizar cambios en el receptor, reemplazando el actual por un radio comercial de mayor utilidad, ampliando así las frecuencias de operación del radiotelescopio construido.

Calibrar los datos de un radiotelescopio es uno de los procesos más importantes para garantizar la veracidad de los mismos. Este proceso involucra la determinación de variables como la temperatura y configuración de ganancia del receptor, además de la verdadera eficiencia del plato reflector, permitiendo definir las capacidades y sensibilidad del instrumento (O'Neil, 2001). Cada radiotelescopio es único, por lo que un sistema de referencia universal que haga equivalente las mediciones realizadas bajo diferentes condiciones deja de ser una alternativa. Como alternativa de solución a esta problemática se recurre a radiofuentes con un flujo de radiación conocido como punto de referencia, llevando a cabo el proceso de calibración a través de métodos observacionales. La temperatura del sistema, crucial en el funcionamiento del radio receptor, es uno de los parámetros con mayor variación en el sistema de recepción, por lo que la complejidad del método utilizado para su determinación puede variar en función de su precisión. Este parámetro se ve influenciado por contribuciones en temperatura de diferentes variables como las condiciones atmosféricas, ruido de fondo cósmico, entre otros. Uno de los métodos más utilizados es el de cargas fría y caliente, el cual asume la respuesta del radio receptor como una medida de temperatura. Este proceso consiste en apuntar el receptor hacia una carga con temperatura conocida (generalmente se implementa un gas o líquido conocido), siendo esta la carga caliente, para posteriormente apuntar el cielo de fondo, siendo esta la carga fría (O'Neil, 2001). Esta técnica permite interpretar la proporcionalidad entre el voltaje de respuesta del sistema y la temperatura medida, haciendo posible la sustracción de la temperatura de ruido aportada por el sistema electrónico del receptor. Aunque este proceso no es realizable para longitudes de onda mayores a 5 cm, para la banda de 12 GHz resulta un método aplicable. La calibración de este parámetro a través del método de carga fría y caliente involucra el uso de una fuente de ruido como punto de referencia, lo que resulta en un incremento de costos a la hora de calibrar el equipo. Aunque una fuente de ruido como punto de referencia favorece la adquisición de un mejor resultado, este elemento puede ser prescindible, como se muestra en Bhatia et al. (2006), donde logran resultados satisfactorios al realizar la calibración de una antena compacta implementando el Sol como fuente de información, gracias a su característica de radio fuente de flujo conocido. Por otro lado, la configuración de ganancia del receptor es determinada a través de observaciones de cielo de fondo conmutando con radiofuentes conocidas. Uno de los métodos más efectivos a

la hora de realizar este procedimiento son los ajustes de referencia, los cuales consisten en amoldar la curva más aproximada al patrón gaussiano de radiación de la antena, permitiendo determinar el ancho del haz, nivel de foco y ganancia real del radiotelescopio (O’Neil, 2001). Aplicar esta metodología no requiere de excepcionales sesiones de adquisición, ya que, como se muestra en R. Hoobi y M. Abood (2019) la captura de un tránsito solar es suficiente para ajustar la curva más aproximada al patrón de radiación de la antena y así determinar los parámetros que esta figura permite interpolar.

Además de la calibración del sistema de recepción, realizar una correcta puesta a punto del sistema de posicionamiento es de suma importancia, ya que, la exactitud de apuntado del instrumento define su utilidad para la observación de objetos celestes. Este proceso de calibración varía en función del tamaño del instrumento. Antenas de gran tamaño hacen necesario el uso de voluminosas estructuras para su sostenimiento, además de robustos sistemas de motores y engranajes para moverla, por lo que un error en el sistema de apuntado puede estar influenciado por innumerables variables. Por otro lado, antenas de menor diámetro con sistemas de movimiento de menor complejidad pueden ser calibradas con procesos observacionales, siendo la posición del Sol un punto de referencia. En el año 2012, una metodología para la calibración del apuntado de radiotelescopios compactos se implementó en el laboratorio de astrofísica de VATLY, Hanoi, Vietman, donde se instaló un radiotelescopio reflector de 2.6 m de diámetro, con movimiento en acimut y altitud, centrado a la observación de la línea de Hidrogeno neutro (1.4 GHz). El proceso realizado involucró la observación solar durante varios días del año, entre octubre y diciembre del 2011, a través de una cuadrícula de observación de 2.6° de ancho compuesta por 25 nodos centrada en el Sol (Hiep et al., 2012). Al generar esta matriz de datos es posible visualizar la desviación entre la verdadera posición del Sol y el nodo central de la cuadrícula, donde se supone está apuntando el radiotelescopio, por lo que la separación angular entre ambos puntos puede ser determinada utilizando métrica euclidiana. Una vez calculada la magnitud del error en cada sesión de observación se define una curva Gaussiana, la cual describe la desviación en la dirección de acimut y altitud del radiotelescopio. En complemento, la cuantificación de la desviación en acimut respecto al norte y en altitud respecto a la horizontal se logra a través del ajuste de la ecuación asociada a cada movimiento, donde las constantes a ajustar representan el nivel de desviación inherente al sistema en cada coordenada (Hiep et al., 2012). Otros métodos para determinar la exactitud del apuntado pueden ser interpolados de sistemas de rastreo satelital, donde el seguimiento del Sol durante un día proporciona la información necesaria para determinar la desviación de apuntado de la antena. En complemento al método anterior esta nueva estrategia además de calcular la desviación espacial en acimut y altitud, permite cuantificar los errores de colimación en ambas coordenadas, además de la inclinación del

pedestal donde descansa el instrumento (Muth et al., 2011). Nuevamente, la detección del Sol a través de mallas de observación permite determinar el error de apuntado en cada una de las direcciones del movimiento. Posterior a esto se ajusta las constantes de la ecuación que relaciona el error en acimut y altitud con la potencia de salida del radiotelescopio a través de la técnica de mínimos cuadrados. Finalmente, se definen 2 ecuaciones de error que relacionan las desviaciones acimutal y de altitud con errores instrumentales través del método Gauss-Helmert (Muth et al., 2011).

El hecho de detectar objetos celestes de baja luminosidad implica un reto para radiotelescopios pequeños, ya que, el tamaño del plato reflector es el factor más influyente en la resolución de la antena. Estos retos han sido superados gracias a los aportes realizados por la radioastronomía amateur, ampliando el catálogo de radiofuentes observadas con radiotelescopios de baja envergadura. Un ejemplo de esto son los resultados obtenidos en Thiruvananthapuram, India, en el año 2016, donde se implementa un radiotelescopio centrado a la recepción de señales a 11.2 GHz con una antena de 0.68 m, un LNB, un buscador de satélites, un Arduino Uno, un sistema de alimentación y una computadora (Mandal et al., 2016). Aunque el sistema presentado no es de gran complejidad, los resultados mostraron que es posible recibir valores del Sol, Saturno y hasta de un brazo de la Vía Láctea, inclusive bajo los impedimentos que presenta realizar el movimiento de forma manual. Por otro lado, en el año 2014, en Senigallia, Italia, se logra la detección del tránsito del remanente de supernova Cassiopeia A a 11 GHz con una antena de 2.3 m. Lograr esta detección involucra valorar el flujo de radiación proveniente del objeto a captar, además de la sincronización del tiempo de entrada y salida del tránsito frente al haz principal de la antena (Falcienelli, 2014). Este procedimiento permite determinar la temperatura de brillo del objeto de interés, además de viabilizar el proceso de calibración a través de múltiples radiofuentes.

Desde el Observatorio Astronómico de la Universidad tecnológica de Pereira se han adelantado estudios en radioastronomía desde el monitoreo del clima espacial en 23.9 kHz (Jaramillo Alvarado et al., 2017), la observación de la actividad solar y joviana en la banda de 20.1 MHz (Galvis Rodriguez et al., 2016) y la observación solar en la banda KU (Higuita et al., 2019). La realización de estos proyectos ha requerido la participación constante de estudiantes de pregrado de la UTP que participen en el grupo de investigación, por lo que la elaboración de una guía enfocada a la enseñanza de la radioastronomía, desde los conceptos básicos hasta entornos prácticos, usando el radiotelescopio en la banda KU es de suma importancia. Este proyecto plantea la calibración y puesta a punto del radiotelescopio de 1.2 m instalado en el año 2018 en el OAUTP centrado a la observación en la banda KU del espectro de radio. Aunque la funcionalidad del equipo se ha corroborado percibiendo un

tránsito solar, la calibración del sistema de recepción y de apuntado abre las posibilidades a la detección de nuevas radiofuentes, ampliando el potencial científico del instrumento y su aplicabilidad en la enseñanza de la radioastronomía.

Capítulo 2

Radiotelescopio en banda KU del OAUTP

2.1 Descripción

En el Observatorio Astronómico de la Universidad Tecnológica de Pereira, en el año 2018, se instaló un radiotelescopio enfocado a la observación en banda KU. Este sistema está compuesto por una antena parabólica de 1.2 m de diámetro en malla de 2.8 mm, la cual descansa sobre una montura altacimutal de alta resolución (ver Figura 2.1). El equipo cuenta con un sistema de recepción basado en tecnología de la televisión satelital con el objetivo de reducir costos y obtener la mejor relación señal-ruido en la respuesta del sistema. En la actualidad el sistema se encuentra enfocado la observación solar y la detección de eventos solares en el rango de 10 - 12.75 GHz.



Figura 2.1: Radiotelescopio OAUTP.

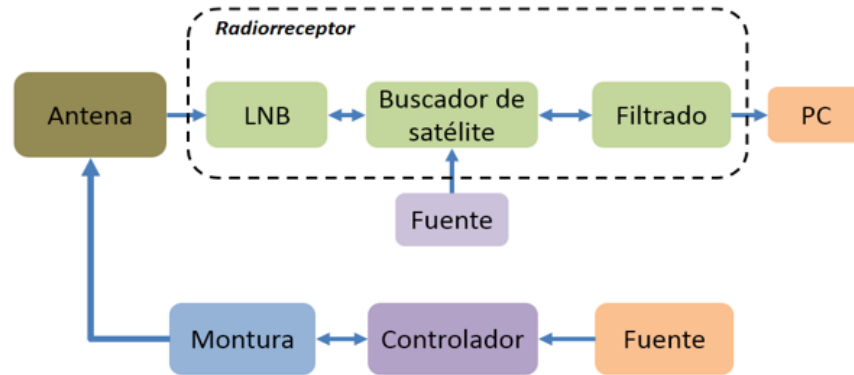


Figura 2.2: Diagrama del radiotelescopio OAUTP.

Los elementos que constituyen el radiorreceptor se muestran en la Figura 2.3. Entre los componentes encontramos un bloque convertidor de bajo nivel de ruido (LNB), por lo que el rango de frecuencias de recepción se ubica entre los 10.7 - 12.75 GHz, mientras que, el rango de frecuencias a la salida del bloque se encuentra entre los 950 - 2150 MHz. Para realizar el tratamiento de la señal obtenida se lleva esta a un buscador de satélites, el cual permite realizar la polarización del LNB además de visualizar la intensidad de la señal a través de un galvanómetro. Una vez obtenida la respuesta en voltaje del buscador de satélites se lleva esta a un filtro pasa bajos con una frecuencia de corte de 500 MHz. Finalmente, la señal es digitalizada a través de una tarjeta de adquisición de datos (DAQ) para luego ser procesada a través de un ordenador. El diagrama de bloques general del radiotelescopio se observa en la Figura 2.2.



Figura 2.3: Radiorreceptor.

Para realizar el movimiento en acimut y altitud, el radiotelescopio cuenta con una montura altacimutal motorizada de alta resolución SPX-AX400 diseñada para trabajar al aire libre, la cual permite hacer movimientos mínimos de 0.2° en ambos ejes. Este sistema cuenta con el controlador SPID MD-02 y la fuente de alimentación SPID PS-02, ambas proporcionadas por el fabricante de la montura (ver Figura 2.4). El montaje hace posible realizar movimientos con el radiotelescopio de forma remota a través de las indicaciones ingresadas al teclado del SPID MD-02, mientras este muestra simultáneamente la posición actual en la pantalla LCD frontal. El controlador utilizado también cuenta con el respectivo software suministrado por el fabricante, permitiendo realizar el envío de coordenadas y la manipulación de parámetros de la montura (ver Figura 2.5).



(a) Controlador y fuente de alimentación.

(b) Montura

Figura 2.4: Sistema de movimiento.**Figura 2.5:** Software de control de la montura.

Desde su instalación el radiotelescopio opera a una frecuencia de 11 GHz, por lo que, dada la geometría de la antena, cuenta con un ancho de haz de 1.9° y una ganancia de 37 dB, permitiendo identificar objetos que se encuentren separados a una distancia angular mínima de 2° . Su correcto funcionamiento se aprecia en su capacidad de registrar un tránsito solar de forma exitosa, logrando además el seguimiento del Sol como se muestra en la Figura 2.6. El sistema alcanza resultados satisfactorios al realizar observación bajo condiciones climáticas adversas, evidenciando su funcionalidad sin presentar limitaciones por cuestiones climatológicas como se muestra en la Figura 2.7. Las características del radiotelescopio se presentan en la Tabla 2.1.

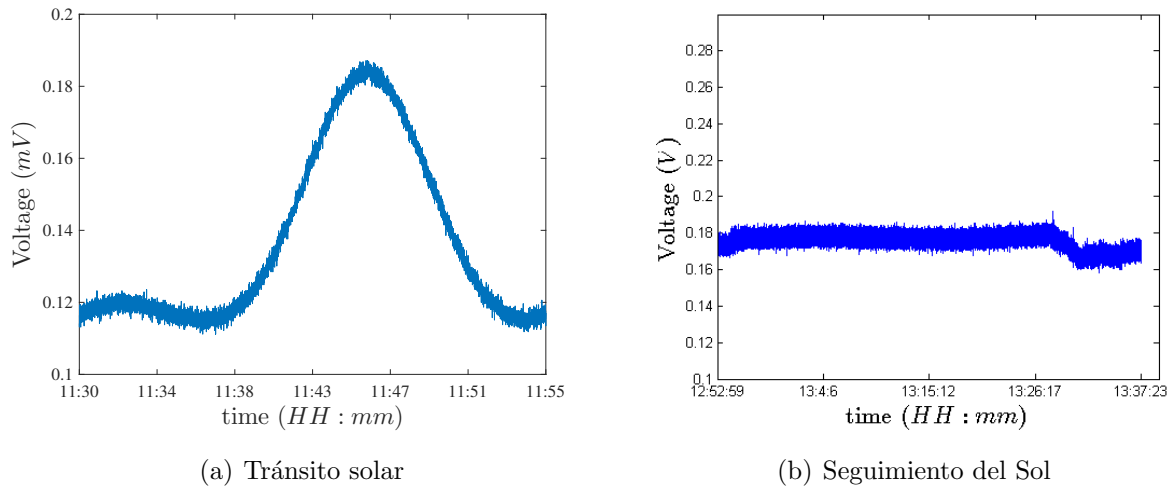


Figura 2.6: Resultados arrojados por el radiotelescopio.

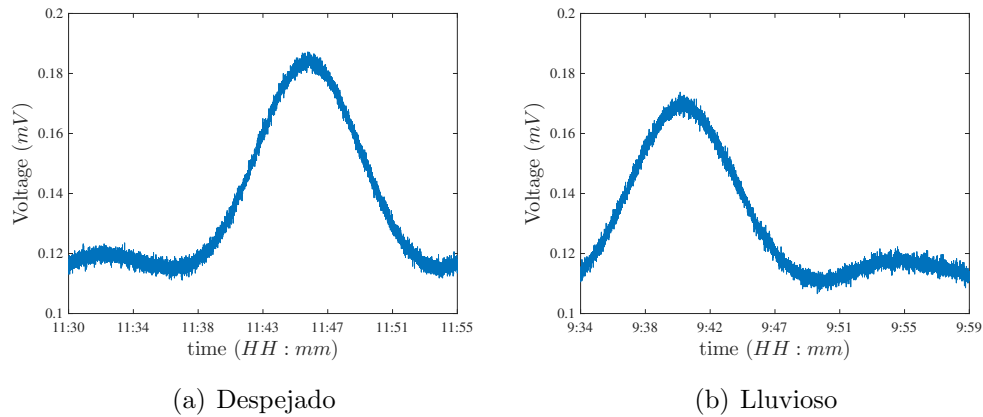


Figura 2.7: Tránsito solar bajo condiciones climáticas diferentes.

Parámetro	Valor	Unidades
Diámetro	1,2	m
Distancia Focal	0,72	m
Cuadrícula de la malla	2,8	mm
Peso	6	Kg
Relación F/D	0,6	N/A
Frecuencia	11	GHz
Ganancia	37,3	dB
Ancho de haz	1,9	°
Rango de movimiento acimut	0 – 360	°
Rango de movimiento altitud	0 – 180	°
Resolución de movimiento	0,2	°

Tabla 2.1: Características del radiotelescopio.

2.2 Puesta a punto

Dado el estado en el cual se encontró el instrumento y debido a los numerosos fallos e imprevistos presentados durante la manipulación y operación del radiotelescopio, tales como la pérdida de comunicación entre el controlador y la montura, la no detección de los motores y bloqueos continuos, y el rebote no deseado de la antena al finalizar su desplazamiento, se optó por realizar algunas mejoras y adaptaciones que permitieran un manejo más cómodo, sencillo y fiable del instrumento.

2.2.1 Cambio de la caja de distribución

Uno de los inconvenientes encontrados durante la manipulación del radiotelescopio fue la pérdida de señal y la no detección de los sensores de posición. En este sentido, se hizo un seguimiento de la señal enviada por los sensores, desde la montura hasta el controlador. Se logró identificar la causa del problema, generado por la oxidación en las conexiones al interior de la caja de distribución que se ubica a un costado del plato parabólico.

En búsqueda de dar una solución, se cambió la antigua caja de distribución por una nueva de mayor tamaño (ver Figura 2.8). Este cambio se realizó con el propósito de separar las conexiones de los sensores que se encontraban ubicados en la zona inferior de la caja, lugar donde se presentaban acumulaciones de agua y por consiguiente la oxidación de las conexiones. Con este cambio, se espera reducir al máximo las interrupciones y pérdidas frecuentes

de la comunicación entre los sensores de la montura y el controlador.

Además, con la nueva caja de mayor tamaño, se lograron ubicar las conexiones de señal y corriente de una forma más apropiada y óptima, como también se facilita el acceso y la manipulación al interior de la misma para cualquier tipo de trabajo de mantenimiento.



(a) Antes.

(b) Después.

Figura 2.8: Caja de distribución.

2.2.2 Actualización del firmware del controlador y modificación en los parámetros para la operación

Desde un principio del desarrollo de este proyecto, el controlador presentó inconvenientes en cuanto a su funcionamiento. Bloqueos durante la operación y la no detección de los motores son algunos de los problemas presentados. Después de una completa revisión del hardware, que incluye la fuente de alimentación, controlador, motores, sensores, conexiones, cableado y fusibles, se logró asegurar el correcto estado de cada uno de los componentes. Por lo tanto se procedió a contactar con el fabricante a través de correo electrónico. Después de efectuar las revisiones sugeridas por el fabricante en respuesta a la solicitud, se procedió a realizar una actualización del firmware como lo recomendaba finalmente el fabricante. La actualización corrigió los problemas que se presentaban.

Uno de los inconvenientes más evidentes que se presentaban durante la operación del radiotelescopio era el rebote realizado por la antena al finalizar cualquier desplazamiento. Este efecto, no deseado, era generado debido a la alta velocidad de desplazamiento imprimida por los motores que trabajaban a máxima potencia. El impacto generado al frenar los motores imposibilitaba la inmediata toma de datos una vez llegase a la posición indicada, requiriendo unos segundos para poder realizar una correcta captura de datos. Este inconveniente fue corregido desde el controlador de la montura al ajustar los parámetros de ambos motores, específicamente al disminuir a un 30% la potencia de operación como se observa en la Figura 2.9. La modificación se refleja en un desplazamiento más lento y suave de la antena. Se encontró que este valor de 30% se encuentra en un rango apropiado para la operación del radiotelescopio ya que permite un desplazamiento suave en ambos ejes y da la posibilidad de recorrer el cielo para llegar a los objetos en un tiempo no muy extenso.

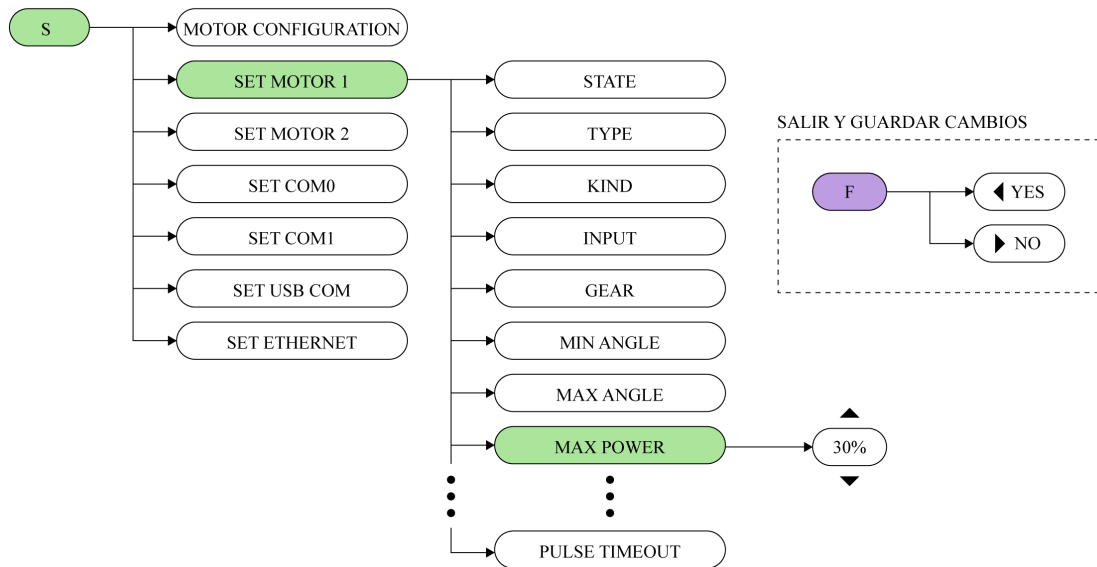
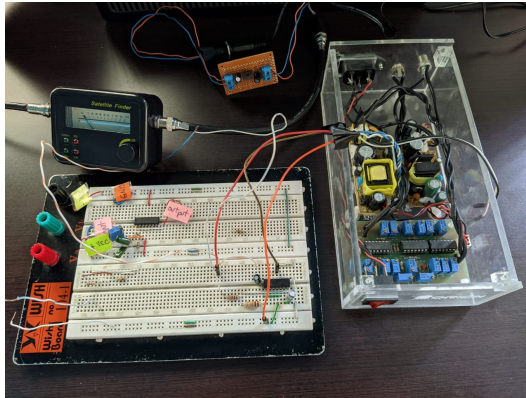


Figura 2.9: Esquema para el ajuste de la potencia de los motores.

2.2.3 Unificación de los circuitos

Dada la complejidad y dificultad que se presentaba a la hora de realizar las conexiones necesarias para la adquisición de datos, se optó por reorganizar los componentes electrónicos en una única caja que solo cuenta con una entrada, una salida y su respectiva fuente de alimentación como se observa en la Figura 2.10. Como se muestra en el esquema circuital de la Figura 2.11, esta caja es polarizada con una fuente dual de 32 V que permite el funcionamiento de cada uno de los componentes, entre ellos el filtro butterworth pasa bajos de segundo orden con frecuencia de corte de 14 Hz, el LNB y el buscador de satélites.



(a) Antes.



(b) Después.

Figura 2.10: Conexiones para la adquisición de datos.

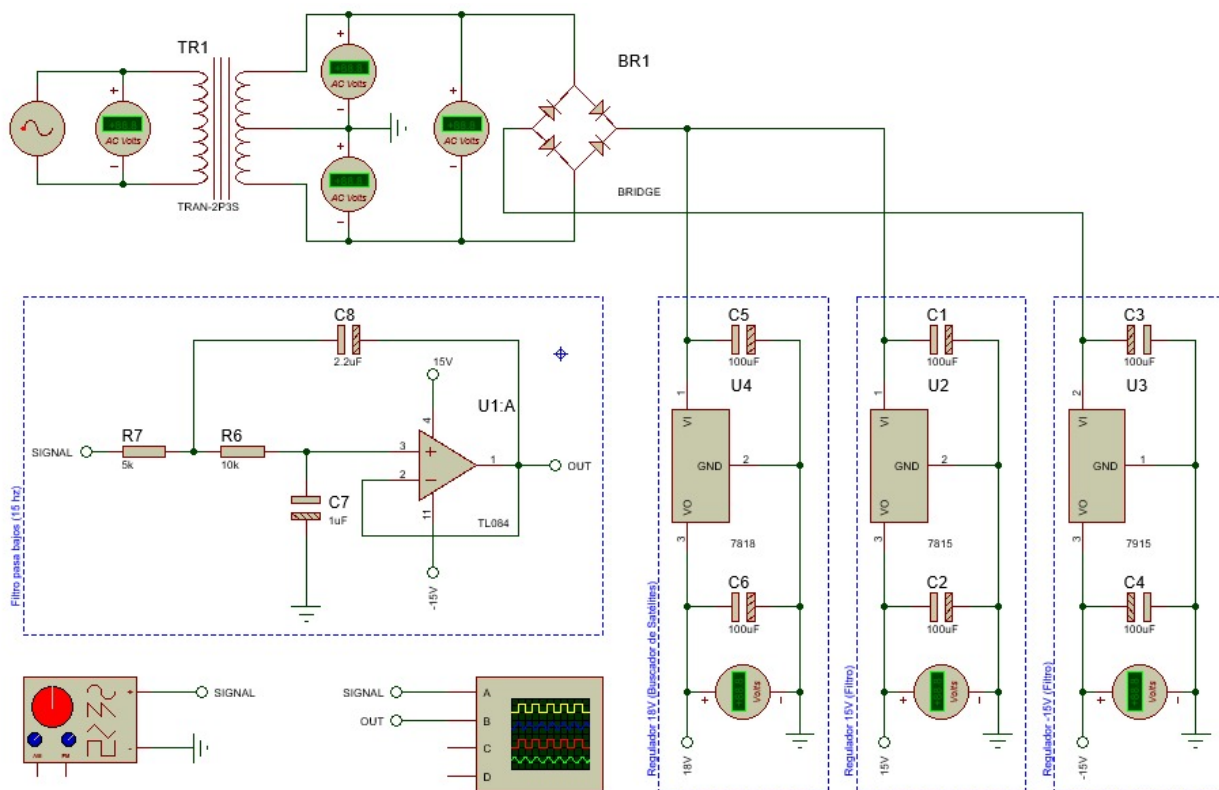


Figura 2.11: Diagrama circuital de la caja de alimentación y conexiones.

Capítulo 3

Calibración

3.1 Calibración del apuntado

La fidelidad en el apuntado de un instrumento es un aspecto de gran relevancia a la hora de realizar investigaciones de carácter científico o académico. Conocer con precisión las capacidades y limitaciones del instrumento permite enmarcar su uso en un grupo más focalizado según los parámetros a estudiar.

En el caso del radiotelescopio del OATUP, los errores presentes en el apuntado tiene gran influencia en la adquisición, análisis y resultados obtenidos. Teniendo en cuenta que los objetos de mayor interés tienen tamaños angulares similares, o menores, a la resolución de desplazamiento de la montura, un error en el apuntado del instrumento podría arrojar datos inconsistentes o erróneos. En este sentido, conocer las desviaciones presentes en las coordenadas de desplazamiento de la antena permitirá corregirlas periódicamente para así garantizar un correcto estudio de los objetos de interés.

3.1.1 Error instrumental

Desde la propia instalación del radiotelescopio del OAUTP hay presente un leve error en la orientación del instrumento debido a la dificultad misma que implica instalar una antena de tal magnitud con un alto grado de precisión. Este error en el apuntado debe ser medido con precisión para después ser corregido. Debido a la dificultad que implicaría una corrección en la instalación y orientación del radiotelescopio, se opta por una corrección que se aplicará al apuntado de la antena a través del controlador de la montura redefiniendo los valores de 0° en acimut y 90° en altitud.

Para poder conocer la desviación en el apuntado, es necesario tener objetos de referencia con posición conocida sobre la bóveda celeste, que además emitan en la banda KU. En este sentido, para la calibración del radiotelescopio se opta por tomar como fuente de radiación conocida al Sol, o los diversos satélites geoestacionarios de televisión satelital. De esta forma tenemos objetos con unos parámetros de referencia a los cuales debemos ajustar nuestro instrumento.

Las posiciones del Sol como también de los satélites geoestacionarios pueden ser consultadas en el software Stellarium. Particularmente, los satélites se pueden encontrar al poner en el buscador del programa “IntelSat” donde el buscador arrojará varias sugerencias de los diversos satélites con esta denominación, algunos de ellos son: intelsat 21, intelsat 30, intelsat 23, entre otros.

3.1.2 Cuantificación del error instrumental en acimut y altitud

Asumiendo un campo del cielo de tamaño $N \times N$ es posible calcular el error en el apuntado del instrumento, solo si el máximo de radiación del objeto escogido es visible dentro del campo observado. Para su cálculo solo es necesario conocer dos posiciones, posición del apuntado de la antena y la posición real del objeto como se aprecia en la Figura 3.1.

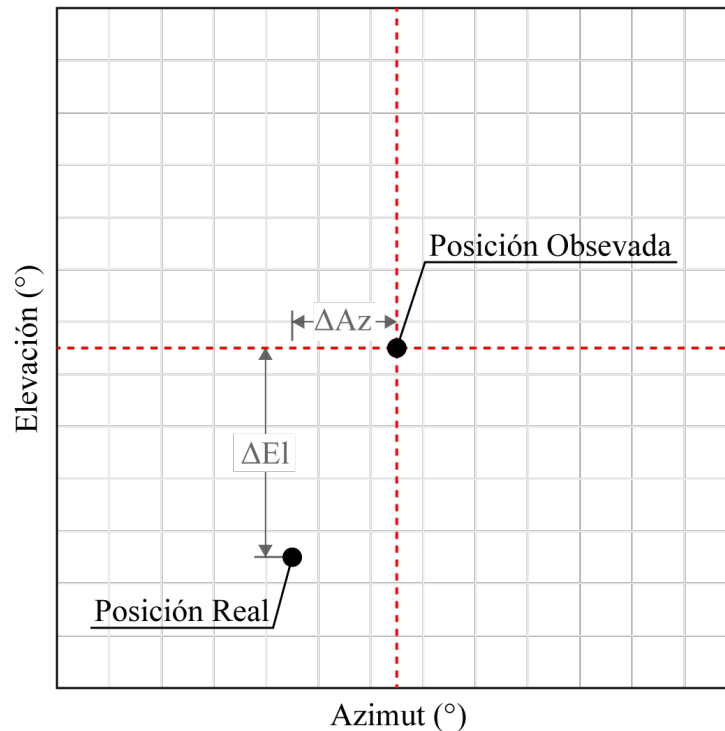


Figura 3.1: Representación gráfica del error en el apuntado.

Según el patrón de distribución de intensidad obtenido del campo observado, es posible calcular la desviación en acimut y altitud teniendo en cuenta las definiciones mostradas en las ecuaciones (3.1.1) y (3.1.2), tomadas de Holleman y Beekhuis (2004).

$$\Delta Az = AzObs - AzReal \quad (3.1.1)$$

$$\Delta El = ElObs - ElReal \quad (3.1.2)$$

Donde ΔAz y ΔEl son las desviaciones en acimut y en altitud, respectivamente; $AzObs$ y $ElObs$ hacen referencia a los ángulos en acimut y altitud del objeto observados por la antena. Finalmente, $AzReal$ y $ElReal$ son los ángulos reales del objeto escogido para la calibración.

Sin embargo, un mayor número de datos permite un mejor análisis de las observaciones, y por consiguiente mejores resultados. De esta forma, el cálculo del error en acimut y en altitud del apuntado de la antena está dado por:

$$ErrorAz = (\Delta Az_1 + \Delta Az_2 + \Delta Az_3 + \dots + \Delta Az_n)/n \quad (3.1.3)$$

$$ErrorEl = (\Delta El_1 + \Delta El_2 + \Delta El_3 + \dots + \Delta El_n)/n \quad (3.1.4)$$

3.1.3 Ventanas de observación

Para obtener las coordenadas de posición del apuntado de la antena y de la posición real del objeto, datos necesarios para la calibración del apuntado del radiotelescopio del OAUTP, se utilizó el apartado de calibración en el software CRT descrito en la sección 4.1.3. Este permite al radiotelescopio realizar ventanas o matrices de observación de tamaño N x N con valores preestablecidos, arrojando el resultado mostrado en la Figura 3.2.

Las ventanas de observación recorren zonas del cielo para así graficar la intensidad de la respuesta de la antena en cada una de las posiciones. De esta forma logramos distinguir en el campo observado la presencia de alguna fuente de radiación que emite con la suficiente intensidad en la banda KU. Estas ventanas ilustran gráficamente el error presente en el apuntado (suponiendo que la desviación de la antena no es mayor al ancho escogido para la ventana de observación).

Teniendo en cuenta que el radiotelescopio fue dirigido hacia la posición real del objeto, esperaríamos obtener una imagen resultante con un máximo de emisión centrado en la ventana de observación. Debido a la desviación de la antena, nos encontraremos con que el objeto en

cuestión se encuentra desplazado con respecto al centro de la imagen, que es la posición real del objeto. Midiendo la desviación en ambos ejes entre el máximo de emisión y el centro de la gráfica podemos estimar una primera aproximación del error presente en el apuntado de la antena, tal como se explicó en la sección anterior.

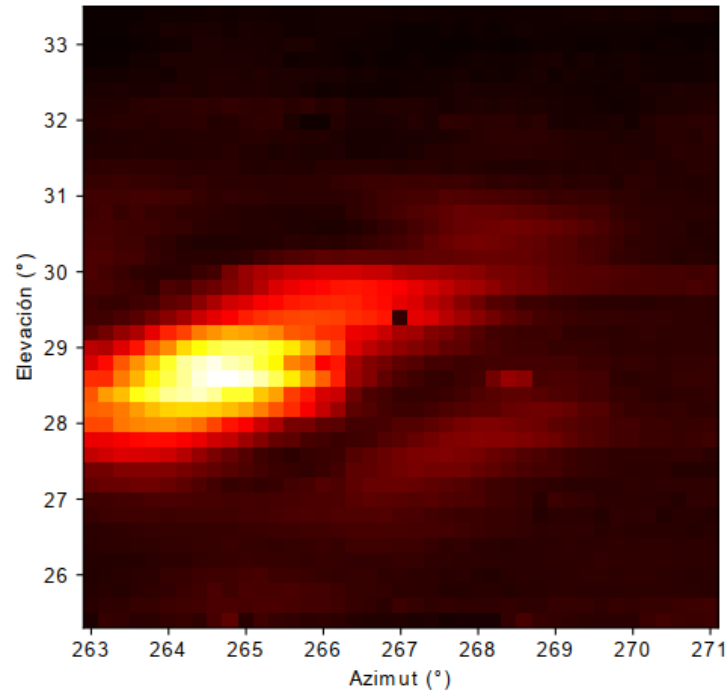


Figura 3.2: Ventana de observación resultante dentro del apartado de calibración.

3.1.4 Calibración en el apuntado del radiotelescopio del OAUTP

El proceso de calibración en el apuntado del radiotelescopio del OAUTP se realizó el día jueves 29 de octubre en horas de la mañana, entre las 10 am y las 11 am, lapso durante el cual el Sol no superó una elevación de 70° . Se recomienda realizar este proceso en horas de la mañana o en horas de la tarde, cuando el objeto con el que se va a calibrar es el Sol. Esto con el propósito de evitar la refracción de la atmósfera presentada a bajas elevaciones, así como también la confluencia de las líneas de acimut en altitudes cercanas a 90° .

La calibración en el apuntado se dividió en dos etapas, una primera de ajuste grueso que permite ubicar el objeto y corregir una primera vez su posición, así se logra enmarcar el máximo de radiación en un campo de menor tamaño para pasar a aumentar la resolución de desplazamiento y disminuir el tamaño de la ventana. Esta etapa es fundamental para asegurar que el objeto escogido sea visible en ventanas de menor tamaño, tal como se ilustra en la Figura 3.3.

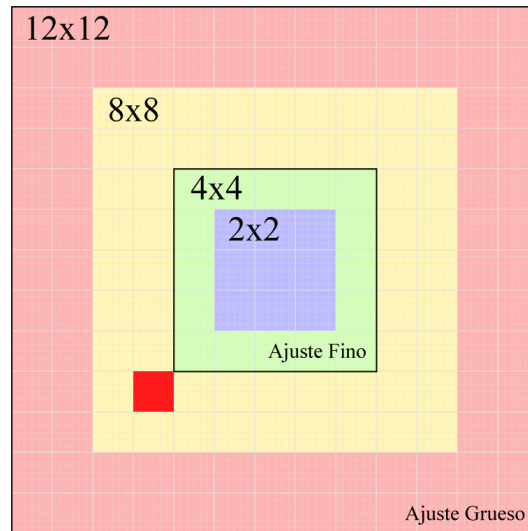


Figura 3.3: Representación gráfica de las etapas de ajuste grueso y ajuste fino.

La segunda etapa, de ajuste fino, permite obtener datos con un mayor grado de precisión en un tiempo aceptable. Si se hiciera una calibración con el mayor tamaño de ventana y la menor resolución se tardaría aproximadamente una hora para recorrer cada una de las 4200 posiciones de dicha ventana.

Primera etapa (Ajuste grueso)

Para el ajuste grueso en el apuntado del radiotelescopio del OAUTP se ingresa al apartado de calibración del Software CRT. En la ventana que se despliega, dentro de la sección de posicionamiento, se introducen las coordenadas del Sol y se ajustan los parámetros para una ventana de observación con un tamaño de $12^\circ \times 12^\circ$ con pasos o resolución de 1° . Una vez rectificada la información se procede a enviar para comenzar con la captura de datos.

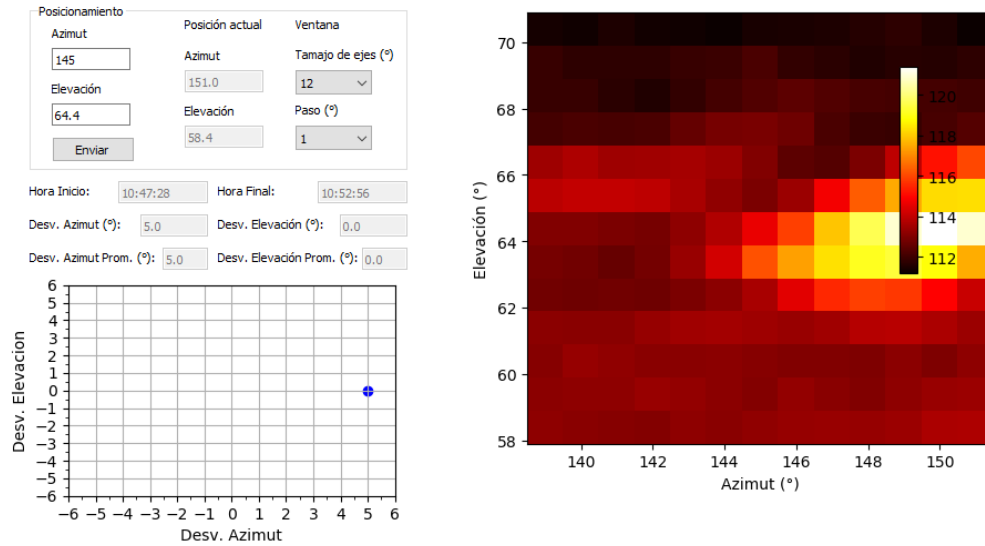


Figura 3.4: Ventana de calibración para el Sol con un tamaño de $12^\circ \times 12^\circ$ y una resolución de 1° . En ella se aprecia el Sol desviado del centro de la imagen.

La Figura 3.4 muestra gráficamente la posición del Sol con respecto a la referencia que es el centro de la ventana, también arroja dos valores de desviación, uno en acimut y uno en altitud, que permite hacer una primera corrección desde el controlador MD-02 en caso que la desviación resultante sea mayor a 1° . Esta corrección se realiza en el modo de calibración del controlador, para ingresar a este modo se presiona el botón F y se procede a modificar los ángulos de ambos motores.

Seguidamente, se realiza una ventana de observación con los mismos parámetros de la primera ventana con el propósito de verificar el ajuste realizado. Una vez corroborado el ajuste se procede con la segunda etapa.

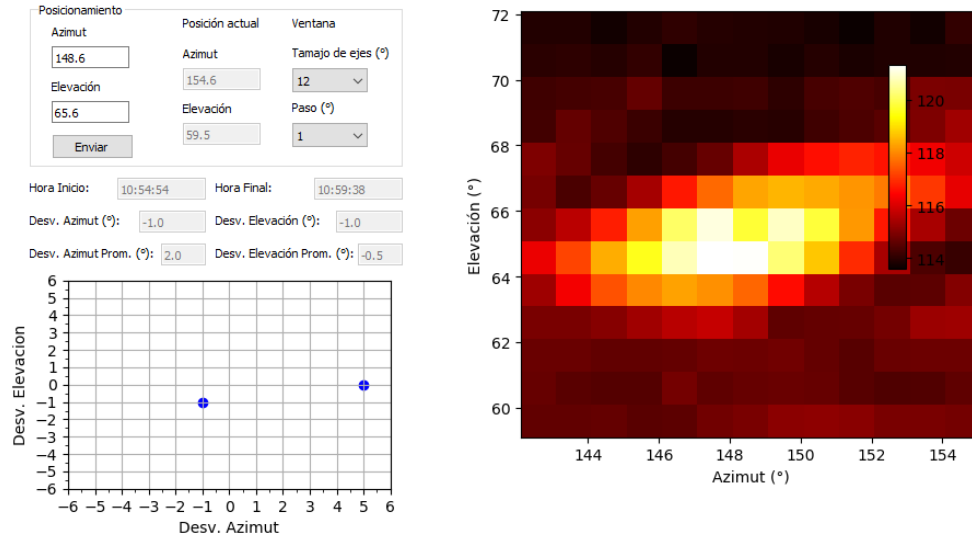


Figura 3.5: Ventana de calibración para el Sol con un tamaño de $12^\circ \times 12^\circ$ y una resolución de 1° . En ella el Sol se encuentra centrado en la imagen

Segunda etapa (Ajuste fino)

El ajuste fino del apuntado de la antena se realizó a través de 6 ventanas de observación (pueden ser más si así se desea) de tamaño $2^\circ \times 2^\circ$ con la mayor resolución permitida que es de 0.2° . Para proceder con la captura de datos es necesario oprimir el botón “Reiniciar” del programa, para limpiar los valores de desviación encontrados en la etapa anterior, de esta forma no se afecta el promedio calculado durante el ajuste fino.

Una vez introducida la posición del Sol se inicia la captura de datos de todas las ventanas de observación, una a una.

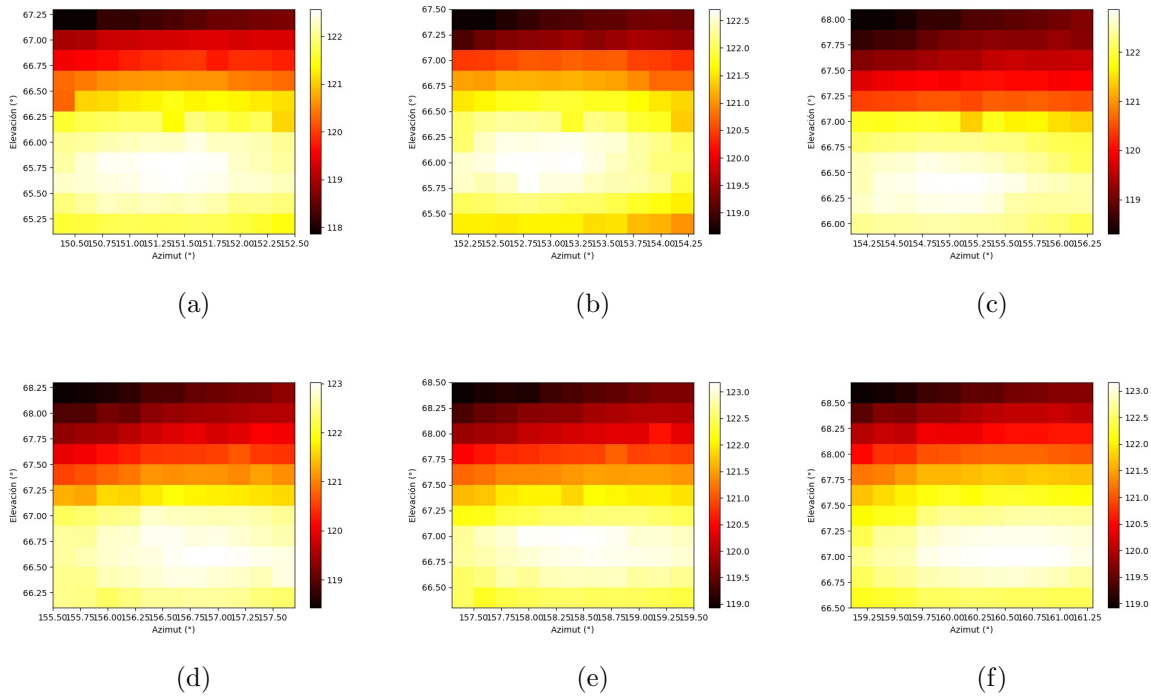


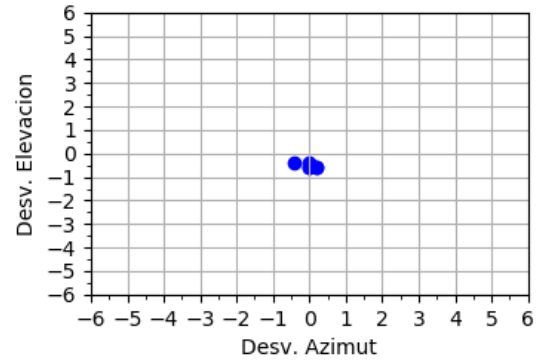
Figura 3.6: Ventanas de calibración para el Sol con un tamaño de $2^\circ \times 2^\circ$ y una resolución de 0.2° .

Las ventanas resultantes evidencian una leve desviación del Sol con respecto al centro de la ventana, error que no era apreciable durante el ajuste grueso debido a la resolución de las ventana, tal como se muestra en la Figura 3.6.

También, como se observa la Figura 3.7, se visualiza de forma gráfica los valores encontrados para la desviación en el apuntado de la antena, estos se ven representados a un costado de la ventana junto con la desviación encontrada en la última ventana y el promedio de todas las anteriores.

Posicionamiento		
Azmut	Posición actual	Ventana
160.2	Azmut	Tamaño de ejes (°)
Elevación	161.2	2
67.6	Elevación	Paso (°)
Enviar	66.5	0.2
Hora Inicio: 11:16:29 Hora Final: 11:18:47		
Desv. Azmut (°): 0.2 Desv. Elevación (°): -0.6		
Desv. Azmut Prom. (°): 0.0 Desv. Elevación Prom. (°): -0.53		

(a) Desviaciones



(b) Gráfica de las desviaciones

Figura 3.7: Desviación encontrada para la ventana actual y promedio de todas las desviaciones.

El software entrega un valor promedio calculado para la desviación en azimut de 0° , y para la elevación de 0.6° tras la aproximación necesaria dada la resolución del instrumento. Conociendo estos datos se procede a realizar el ajuste desde el controlador. Tanto el ajuste fino como la calibración finalizan con la ventana de observación mostrada en la Figura 3.8, la cual permite corroborar el correcto apuntado del radiotelescopio.

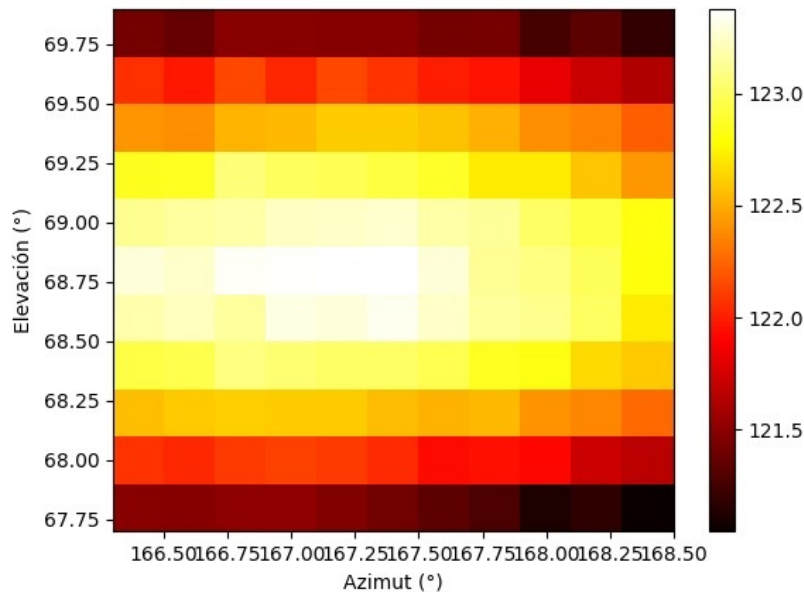


Figura 3.8: Ventana de calibración para el Sol con un tamaño de $2^\circ \times 2^\circ$ y una resolución de 0.2° , en ella se observa el correcto apuntado de la antena.

Tras la calibración en el apuntado del instrumento se hizo notorio un error presente a la hora de observar objetos ubicados en el hemisferio opuesto al usado durante la calibración. Este error puede ser apreciado y medido al apuntar un objeto que se encuentre en dicha región opuesta. La naturaleza de este error es el desnivel presente en la base de la montura, por lo cual, al realizar un cambio de hemisferio, se hace notorio un desfase de dos veces el desnivel del instrumento, tal como se aprecia en la Figura 3.9.

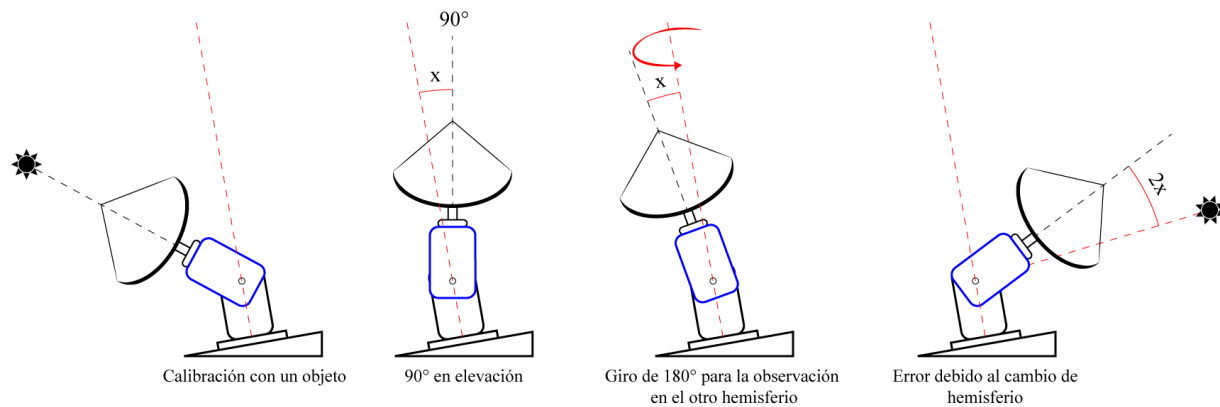


Figura 3.9: Error en el apuntado al realizar cambio de hemisferio.

Luego de varias observaciones en ambos hemisferios (este y oeste) se logró medir este desfase que únicamente se presenta en el eje de elevación, tal desfase es de 3° . Con este valor se determinó que el desnivel presente en el instrumento es de 1.5° . Tras la calibración del radiotelescopio se abren dos escenarios. El primer escenario corresponde a la observación de cuerpos celestes en el hemisferio donde se calibró el instrumento. En este caso puede realizar observaciones con total certeza de que la antena se encuentra apuntando de forma correcta. El segundo caso corresponde a la observación de cuerpos celestes en el hemisferio opuesto al usado durante la calibración. En este caso se hace presente el error generado debido al desnivel de la antena por lo cual se recomienda añadir dicho error (sumar 3° en elevación) únicamente a las observaciones que no requieran un registro de posición altamente precisos. Otra solución, y la más eficaz, es calibrar el apuntado del radiotelescopio en este hemisferio. Una vez conocido este error es posible tener presente las siguientes consideraciones

- Calibrar con cualquiera de los satélites geoestacionarios que emiten en la banda KU cuando el Sol se encuentre a elevaciones no favorables, como también cuando no sea visible.
- Tener presente el hemisferio en el cual se calibró el instrumento para corregir el error a la hora de realizar observación en el hemisferio opuesto. Este error deberá ser añadido a la posición real del objeto para ser introducido en el software.

- Realizar la corrección de este error por medio del Software (trabajo futuro).

3.2 Calibración del plato reflector

3.2.1 Temperatura de operación

Para determinar la temperatura de operación del sistema receptor de forma experimental es necesario realizar una adquisición de datos dividida en dos etapas como se muestra en Gaylard (2012). El primer momento consiste en apuntar una zona donde no haya ninguna radiofuente, en este caso el espacio vacío, donde el voltaje de respuesta representará la lectura de las temperaturas de ruido del sistema (T_{sys}) y cielo de fondo (T_{cold}). Para llevar a cabo este proceso, se caracterizó la respuesta de la antena al apuntar el cielo a 90 ° de altitud. Una vez obtenida la respuesta en voltaje ante las diferentes temperaturas, se define la temperatura del cielo de fondo con una magnitud de 20 K (Gaylard, 2012).

$$\begin{aligned} T_{sys} + T_{cold} &= C \cdot V_1 \\ T_{sys} + 20K &= C \cdot 0.042V \end{aligned} \tag{3.2.1}$$

El segundo momento se basa en apuntar hacia un objeto que cubra todo el ancho de haz de la antena y cuya temperatura sea conocida. Aquí es posible definir el voltaje sensado como la medición de la temperatura de referencia (T_{ref}) junto con el ruido del sistema. Además, se utiliza un pirómetro infrarrojo para determinar la temperatura superficial del objeto de referencia, la cual es de 301.75 K.

$$\begin{aligned} T_{sys} + T_{ref} &= C \cdot V_2 \\ T_{sys} + 301.75K &= C \cdot 0.093V \end{aligned} \tag{3.2.2}$$

Resolviendo el sistema de ecuaciones generado por las ecuaciones (3.2.1) y (3.2.2) se logra determinar la constante de proporcionalidad C , la cual relaciona el comportamiento entre el voltaje de respuesta y la temperatura medida, además que permite el cálculo de la temperatura del sistema.

$$\begin{aligned} C &= \frac{T_{cold} - T_{ref}}{V_1 - V_2} \\ C &= \frac{20K - 301.75K}{0.042V - 0.093V} \\ C &= 5524.5 \frac{K}{V} \end{aligned}$$

Una vez cuantificada la constante de proporcionalidad C , se determina la temperatura del sistema como:

$$T_{sys} = C \cdot V_2 - T_{ref}$$

$$T_{sys} = 5524.5 \frac{K}{V} \cdot 0.042V - 20K$$

$$T_{sys} = 212.03K$$

El procedimiento se aplicó durante varios días, recopilando la temperatura del sistema y la constante de proporcionalidad entre el voltaje de respuesta y la temperatura sensada. En la Tabla 3.1, se observa la variación de la temperatura del sistema durante el transcurso de la tarde. Por otro lado, la Tabla 3.2 permite visualizar el cambio de esta variable durante la noche.

Fecha	Hora	Temperatura del sistema (K)	Constante C
29/10/2020	16:04	178.4	5352.4
	16:37	215.7	5552.6
30/10/2020	13:37	224.9	5646.0
	14:11	214.9	5677.6
	14:45	233.4	5678.6
	15:20	238.4	5583.6
03/11/2020	13:55	140.5	5164.0
	14:33	178.8	5983.2
	15:07	196.9	6204.3
	16:14	219.5	6018.9

Tabla 3.1: Recopilación de la temperatura del sistema durante el día.

Una vez recopiladas ambas variables durante distintos momentos, es posible definir una magnitud promedio para la temperatura del sistema de recepción y la constante de proporcionalidad entre la temperatura observada y el voltaje en la respuesta durante el día.

$$T_{sys}[K] = 204.3 \pm 27.2$$

$$C[K/V] = 5686.8 \pm 427.5$$

Fecha	Hora	Temperatura del sistema (K)	Constante C
29/10/2020	20:14	166.4	5059.4
	21:22	89.3	5200.3
4/11/2020	18:58	121.7	5613.6
	19:24	118.7	5525.6
	19:57	99.1	5136.2
	20:23	75.5	5258.1
	20:50	112.3	5564.3
	21:12	102.1	5505.7
	21:33	92.6	5423.8

Tabla 3.2: Recopilación de la temperatura del sistema durante la noche.

Ahora, se calcula la magnitud de la temperatura del sistema durante la noche, donde se hace notoria la disminución de la temperatura en el ambiente desde que el Sol se ha ocultado.

$$T_{sys}[K] = 108.6 \pm 45.2$$

$$C[K/V] = 5365.2 \pm 277.1$$

3.2.2 Ancho de Haz

El ancho medio de haz de una antena reflectora (HPBW) representa el ancho del objeto que podremos observar con ella, además del ángulo mínimo de separación entre dos objetos para poder ser identificados por la antena individualmente. Una de las formas más comunes de obtener este valor es a través de un tránsito solar. Esta técnica consiste en apuntar la antena hacia una posición por la que pasará el Sol, realizando la adquisición de datos mientras este pasa a velocidad sideral por el campo de visión del instrumento. Una vez finalizada la sesión de adquisición, el ancho de haz puede ser extraído de la gráfica resultante, ubicando el tiempo que tarda la señal en descender 3dB desde su valor máximo. El doble del ancho medio de haz equivale al ancho de haz completo (FPBW) como se ilustra en la Figura 3.10.

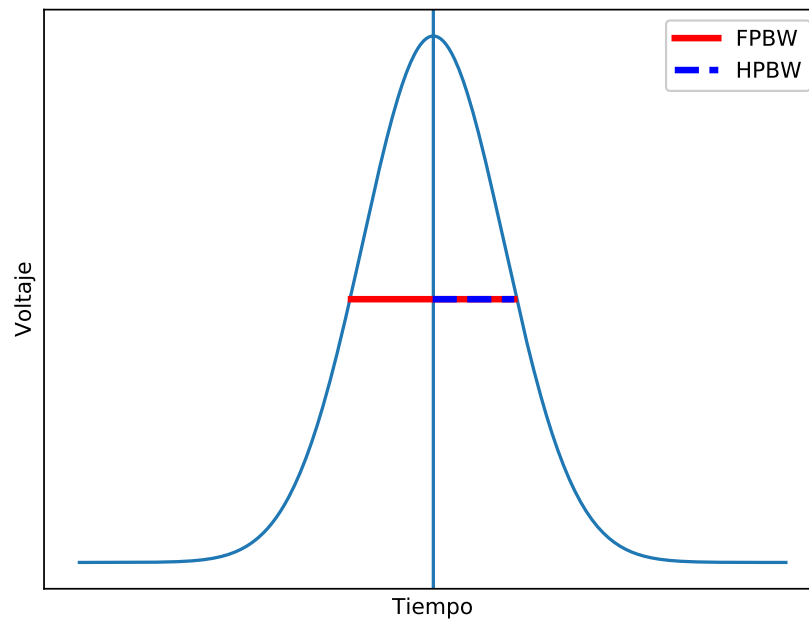


Figura 3.10: Patrón de radiación de la antena y ancho de haz.

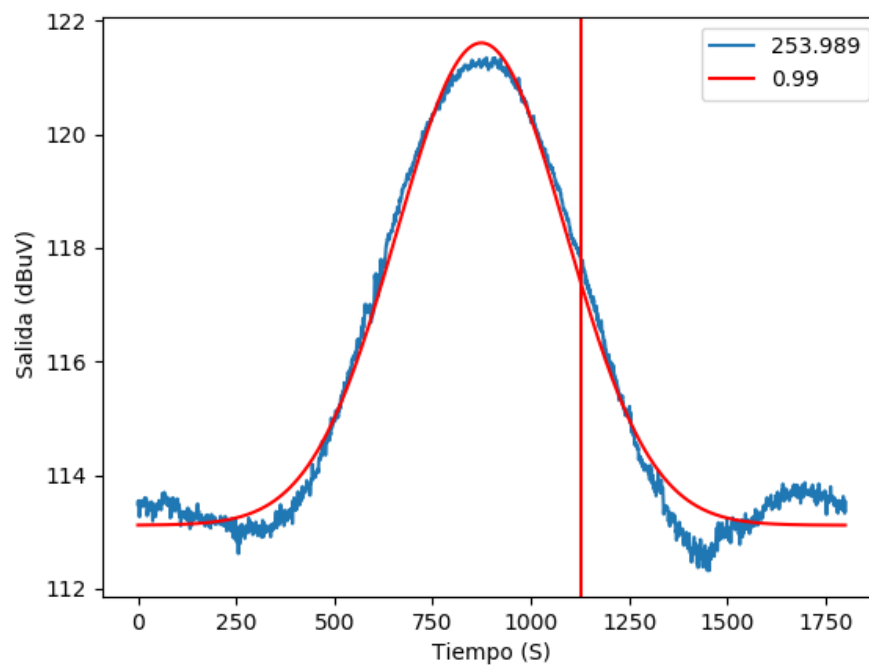


Figura 3.11: Tránsito solar y ajuste Gaussiano.

Tras realizar la adquisición de un tránsito solar como se muestra en la Figura 3.11, se estima un HPBW de 253.98 s. Sabiendo que el Sol se mueve a una velocidad angular de 1° cada 4 minutos, por el coseno de la declinación, se calcula el HPBW en grados de la siguiente manera.

$$HPBW = 253.98s \frac{1^\circ}{240S} \cos(-12.44^\circ)$$

$$HBPW = 1.03^\circ$$

Ahora, se aplica la corrección del HPBW al obtener su valor de convolución, donde θ_A representa el ancho de haz corregido, θ_m el HPBW obtenido, y ω el equivalente al ancho angular del Sol.

$$\theta_A = \sqrt{\theta_m^2 - \frac{\ln(2)}{2}\omega^2}$$

$$\theta_A = \sqrt{1.03^{\circ 2} - \frac{\ln(2)}{2}0.5^{\circ 2}}$$

$$\theta_A = 0.99^\circ$$

Para obtener resultados más sólidos, se realizó el cálculo del HPBW de la antena durante varios días y a diferentes horas, obteniendo los datos mostrados en la Tabla 3.3.

Fecha	Hora	Ancho medio de haz ($^\circ$)
29/10/2020	16:04	1.34
	16:37	1.31
30/10/2020	13:37	0.99
	14:11	1.07
	14:45	1.12
	15:20	1.21
03/11/2020	13:55	1.14
	14:33	1.16
	15:07	1.21
	16:14	1.37

Tabla 3.3: Variación del ancho de haz.

Como se observa en la Tabla 3.3, a medida que la altitud del objeto observado disminuye, el ancho de haz aumenta su tamaño. Para evitar inconvenientes con este valor se recomienda apuntar objetos con una altitud mayor a 20° . Cumpliendo estas condiciones, el ancho medio de haz de la antena y su ángulo sólido ($\Omega_A = \pi\theta_A^2$) son:

$$\begin{aligned}
HPBW[^\circ] &= 1.1 \pm 0.1 \\
HBPW[rad] &= 0.019 \pm 0.002 \\
\Omega_A[sr] &= (1.1 \pm 0.2) * 10^{-3}
\end{aligned}$$

3.2.3 Area efectiva, Eficiencia y Ganancia

Para la estimación de la área efectiva (A_{eff}) del plato parabólico de forma experimental, partiremos del ángulo sólido Ω_A de la antena, obtenido a partir del ancho de haz. Una vez se ha aplicado esta relación, se determina el área efectiva para la frecuencia observada a partir de la relación descrita en la ecuación (3.2.3), obtenida de Chanka (2018), donde se relaciona la longitud de onda central y el ángulo sólido del plato.

$$\begin{aligned}
A_{eff} &= \frac{\lambda^2}{\Omega} \\
A_{eff} &= \frac{(0.02676m)^2}{1.1 * 10^{-3}sr} \\
A_{eff} &= 0.65m^2
\end{aligned}$$

La eficiencia, en una antena parabólica, determina la relación entre su área física y la verdadera área iluminada de esta, como se muestra en la ecuación (3.2.3). Esta variable varía entre 0 y 1, siendo determinante en la ganancia de la antena, ya que está directamente relacionada con la intensidad de señal captada por esta.

$$\begin{aligned}
\eta &= \frac{A_{eff}}{A} \\
\eta &= \frac{0.65m^2}{\pi \cdot (0.6m)^2} \\
\eta &= 0.57
\end{aligned}$$

Finalmente, se calcula la ganancia de la antena como:

$$\begin{aligned}
G &= 10 \log \left(\frac{4\pi A_{eff}}{\lambda^2} \right) \\
G &= 10 \log \left(\frac{4\pi 0.65m^2}{(0.02676m)^2} \right) \\
G &= 40.6dB
\end{aligned}$$

3.2.4 Sensibilidad aproximada

Por razones de costo se efectúa la calibración de la antena de plato reflector sin una referencia de ruido local. En consecuencia el proceso se llevó a cabo siguiendo el procedimiento de cargas frías y calientes como se muestra en Uwe y Frank (2005). Ya que el voltaje de recepción se presenta en $dB\mu V$, asumiremos la resistencia de referencia de 7.5Ω , característica del cable coaxial RG-59/U utilizado en la transmisión de la señal captada desde el LNB.

$$U[\mu V] = 10^{\frac{U[dB\mu V]}{20}}$$

$$P[W] = \frac{U^2}{R}$$

Una vez definida la magnitud de la intensidad de señal recibida, se procede a establecer la relación de ganancia que la antena de plato reflector presenta al apuntar a una radiofuente. Para esto se fija la potencia al apuntar al espacio vacío como P_0 , y la potencia al apuntar una radiofuente como P . El cociente entre ambas magnitudes será el cociente Y , como se ilustra en la ecuación (3.2.3).

$$Y = \frac{P}{P_0} = 10^{\frac{U_{dB} - U_0 dB}{10}} \quad (3.2.3)$$

El factor Y permite relacionar la respuesta del radio telescopio con la temperatura medida por este. Es por esto que se debe tener en cuenta que, al realizar cualquier medición con el radiorreceptor, siempre se encontrará presente en esta la temperatura de ruido generada por el sistema (T_{sys}). Por otro lado, para el espacio vacío se estima una temperatura de 20 K a 12 GHz (T_{cold}), teniendo en cuenta los aportes realizados por la atmosfera terrestre. Finalmente, se fija la temperatura medida por la antena (T_s) como la temperatura de la radiofuente observada.

$$Y = \frac{T_{sys} + T_s}{T_{sys} + T_{cold}} \quad (3.2.4)$$

Teniendo en cuenta que la temperatura ambiente ronda los 300 K, se asume que la T_{sys} es mucho mayor a T_{cold} . En consecuencia la ecuación (3.2.4) se puede reducir de la siguiente forma.

$$Y = \frac{T_{sys} + T_s}{T_{sys}} \quad (3.2.5)$$

Una vez definida la relación entre la temperatura del sistema y la temperatura observada, se pasó a determinar la temperatura de la antena a partir de las ecuaciones que rigen el funcionamiento de las antenas de plato reflector. Para esto es necesario tener en cuenta el

área efectiva de la antena (A_{eff}), el flujo de la radiofuente observada (S) y la constante de boltzman (K).

$$T_s = \frac{S \cdot A_{eff}}{K} \quad (3.2.6)$$

Reemplazando la ecuación (3.2.6) en la ecuación (3.2.5) e igualando esta a la ecuación (3.2.3), se obtiene:

$$\frac{P}{P_0} = \frac{T_{sys} + \frac{S}{K} A_{eff}}{T_{sys}} \quad (3.2.7)$$

$$\frac{P}{P_0} = 1 + S \frac{A_{eff}}{K \cdot T_{sys}}$$

Para finalizar, se aplica logaritmo base 10 a ambos lados, obteniendo el cambio en respuesta de la antena en $dB\mu V$ al apuntar una radiofuente con flujo conocido en Jy

$$UdB - U_0dB = 10 \log \left(1 + S \frac{A_{eff}}{k \cdot T_{sys}} \right)$$

Extrayendo el flujo solar durante el día de observación de la página web de The Learmonth Observatory¹ se estima la respuesta de la antena al puntar al Sol. Por otro lado, se extrae el flujo para los objetos de espacio profundo de la Base de Datos Extragaláctica de la Nasa², para así determinar la magnitud del estímulo generado por estos en el receptor.

$$UdB - U_0dB = 10 \log \left(1 + S \frac{0.65m^2}{k \cdot 204.3K} \right) \quad (3.2.8)$$

Aplicando la ecuación 3.2.8, con el objetivo de predecir el incremento en la respuesta del radiotelescopio al apuntar las radiofuentes enumeradas en la Tabla 5.2, se construye la gráfica de la Figura 3.12. Allí se observa la respuesta del radorreceptor en $dB\mu V$ en el eje vertical frente la magnitud de flujo observada en el eje horizontal.

¹<http://www.sws.bom.gov.au/Solar/3/4/2>

²<http://ned.ipac.caltech.edu/>

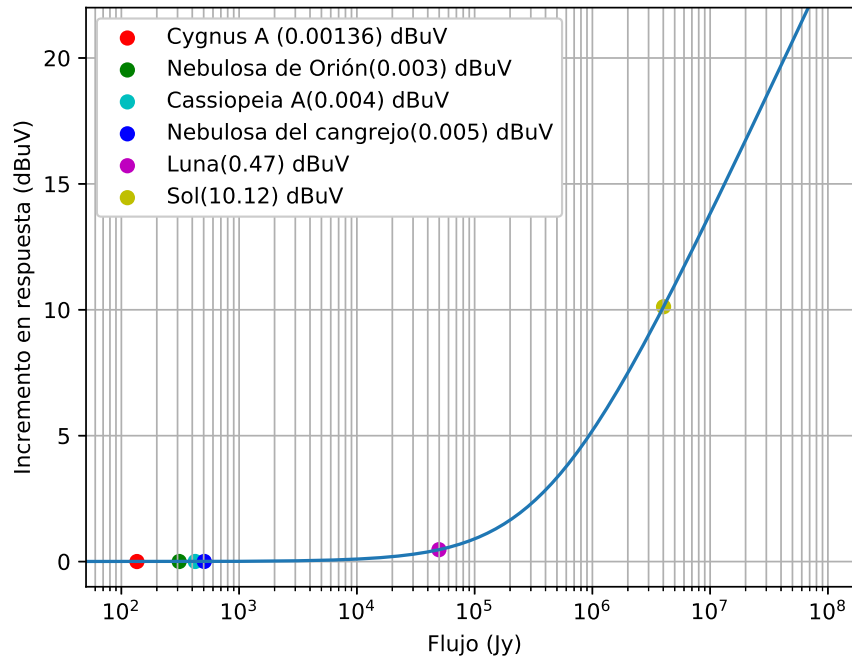


Figura 3.12: Respuesta del radiotelescopio ante diferentes radiofuentes.

Finalmente, tras concluir el proceso de calibración, se enumeran los parámetros resultantes en la Tabla 3.4.

Parámetro	Valor	Unidades
Ancho de haz	2.2	°
Área efectiva	0.65	m^2
Eficiencia	0.57	N/A
Ganancia	40.6	dB
Temperatura del sistema	204.3	K

Tabla 3.4: Parámetros del radiotelescopio.

Capítulo 4

Software

Para realizar la adquisición de datos, además del movimiento en acimut y altitud, se desarrolló una aplicación en el lenguaje de programación Python, implementando la herramienta Qt, facilitando la construcción de la interfaz gráfica de usuario (GUI). A través de esta interfaz es posible realizar la calibración tanto del sistema receptor como del sistema de movimiento. Además, permite escanear zonas específicas del cielo para obtener así una imagen a partir de lo sensado por el radiotelescopio. El diagrama de bloques de la Figura 4.1, ilustra los componentes del software.

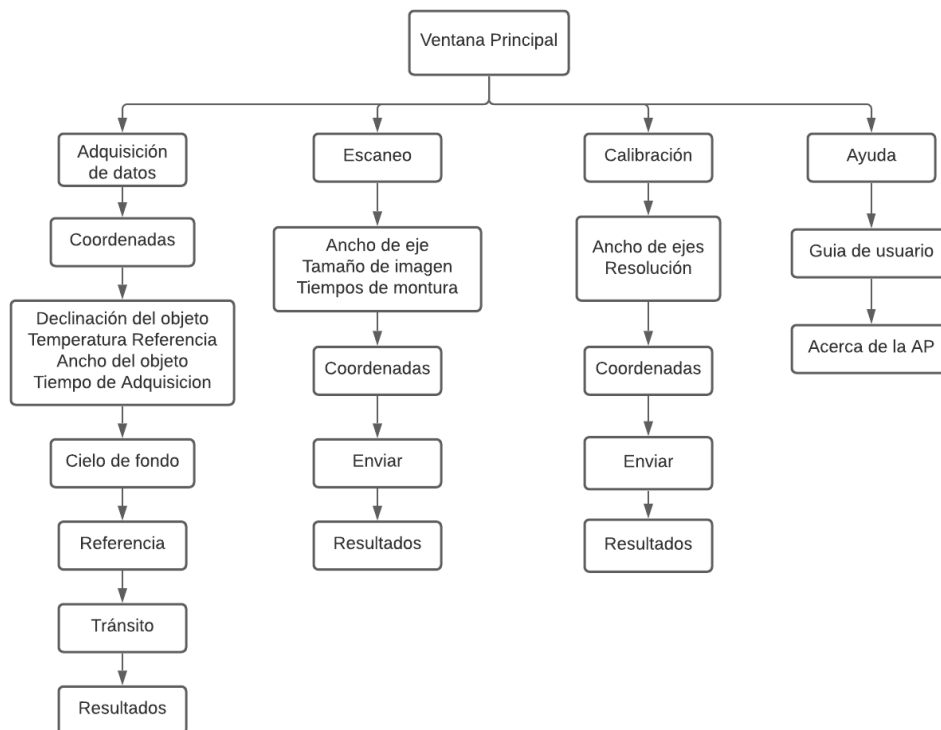


Figura 4.1: Estructura *Compact Radio Telescope UTP*.

4.1 Interfaz gráfica de usuario

4.1.1 Ventana Adquisición de datos

La primera ventana de la aplicación es la nombrada como “Adquisición de datos”, mostrada en la Figura 4.2. Esta, se encuentra involucrada en la determinación de la temperatura de brillo de objetos celestes y en la calibración del sistema receptor, donde es imperativo realizar la adquisición de datos durante tres momentos, apuntando en direcciones diferentes. Allí podremos encontrar cada uno de los botones y entradas de texto necesarios para la ejecución del proceso involucrado en la parametrización del sistema y la observación de cuerpos celestes a través de tránsitos con tiempo definido.

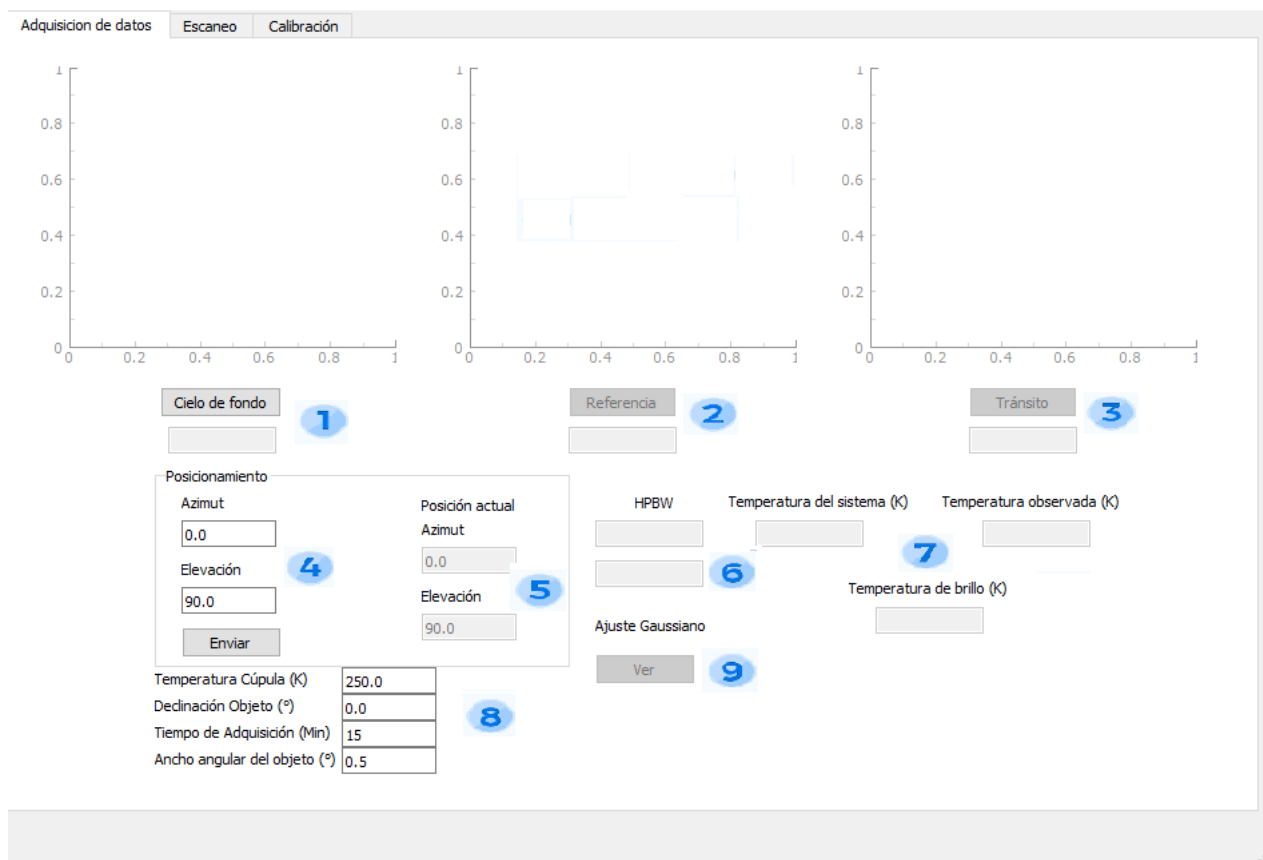


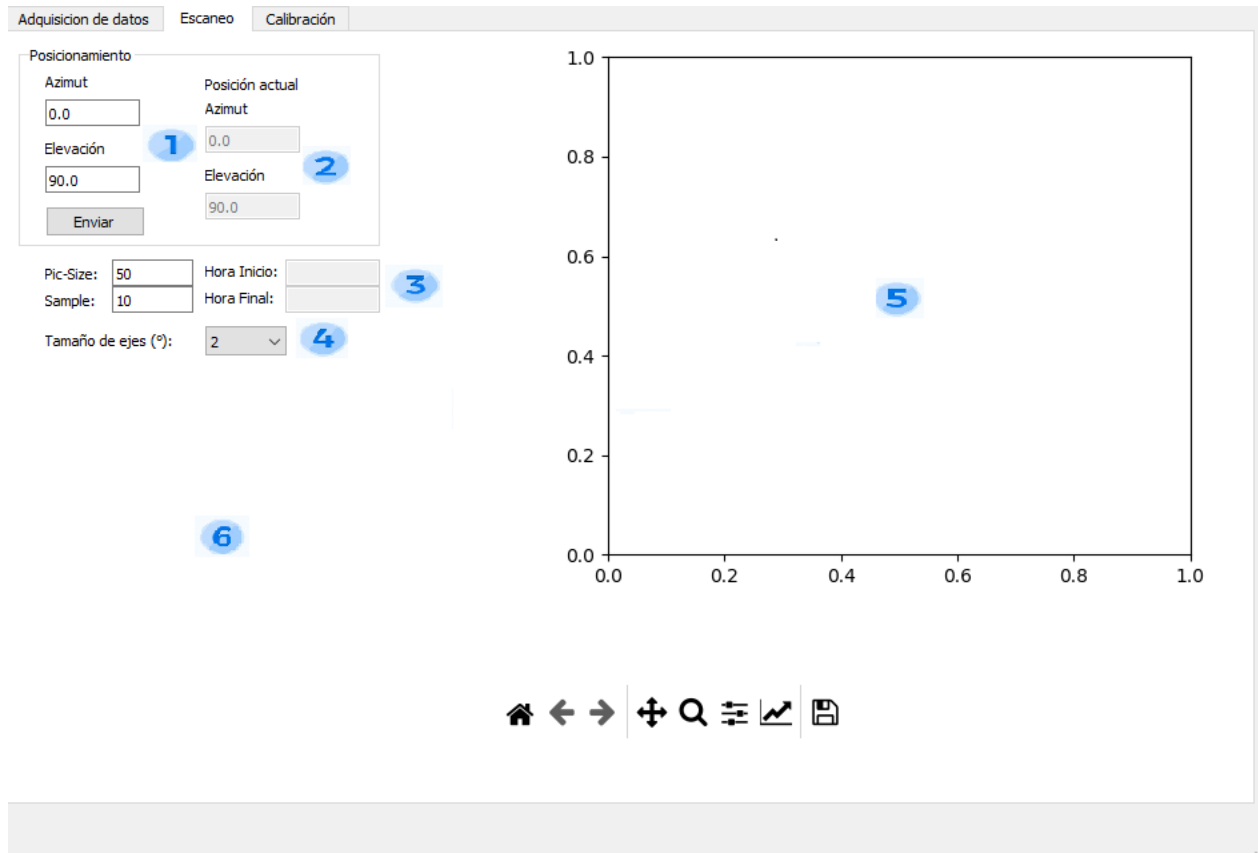
Figura 4.2: Página: Adquisición de datos.

- 1 Ejecuta y grafica la adquisición de datos durante 5 s al apuntar el cielo de fondo.
- 2 Ejecuta y grafica la adquisición de datos durante 5 s al apuntar la referencia.
- 3 Ejecuta y grafica la adquisición de datos durante el Tiempo de Adquisición ingresado.
- 4 Edición de coordenadas a enviar.

- 5 Visualización de la posición actual del radiotelescopio.
- 6 Visualización del Ancho de Haz en segundos y grados una vez finalizado el tránsito.
- 7 Visualización de la Temperatura del Sistema, Temperatura Observada por el radiotelescopio y Temperatura de Brillo del objeto en cuestión.
- 8 Ingreso de los parámetros necesarios para los cálculos finales.
- 9 Visualización del ajuste Gaussiano y el ancho de haz calculado.

4.1.2 Ventana Escaneo

La segunda ventana que podremos encontrar en la aplicación es la nombrada como “Escaneo”. El objetivo principal de esta sección de la aplicación se centra la observación de un objeto celeste a través de escaneos de deriva en una zona específica del cielo. El procedimiento se realiza manteniendo una de las coordenadas de acimut o altitud en un valor constante, mientras se mueve la otra de forma oscilatoria en un rango de grados predeterminado y un tiempo definido. De esta forma, se espera detectar el objeto cuando este pase por el haz principal de la antena. Los botones y entradas de texto de esta ventana, junto con las gráficas resultantes a la salida se muestra en la Figura 4.3.

**Figura 4.3:** Página 2: Escaneo.

- 1 Ingresar coordenada central de la imagen.
- 2 Visualización de la posición actual del radiotelescopio.
- 3 Ingreso del tamaño de la imagen en ambos ejes y visualización de la hora de inicio y finalización.
- 4 Ingreso del ancho angular de la imagen.
- 5 Visualización gráfica de la zona observada durante el período de tiempo elegido.
- 6 Visualización de la imagen .jpg de la zona observada.

4.1.3 Ventana Calibración

La ventana “Calibración” se centra en la observación de una zona específica del cielo a través de una matriz cuadrada de tamaño definible. En la imagen resultante a la derecha cada punto representa un valor de intensidad medido, según la posición en la que se encuentra apuntado el radiotelescopio. Además, a partir de los resultados obtenidos, se calcula la desviación del apuntado del radiotelescopio, asumiendo la coordenada central como el punto donde se ubica el objeto de interés. Para la visualización gráfica de esta sección se construyó la ventana mostrada en la Figura 4.4. Allí se puede encontrar cada uno de los botones y entradas de texto necesarios para la ejecución de la ventana de observación. Entre sus componentes se encuentra el ingreso de la coordenada central, la visualización gráfica del mapa de colores en representación de las intensidades y la gráfica del acumulado de desviaciones. Además, durante cada sesión se muestra la desviación obtenida, la cual es guardada y sumada con la anterior, para así dar en la salida la desviación promedio en cada dirección.

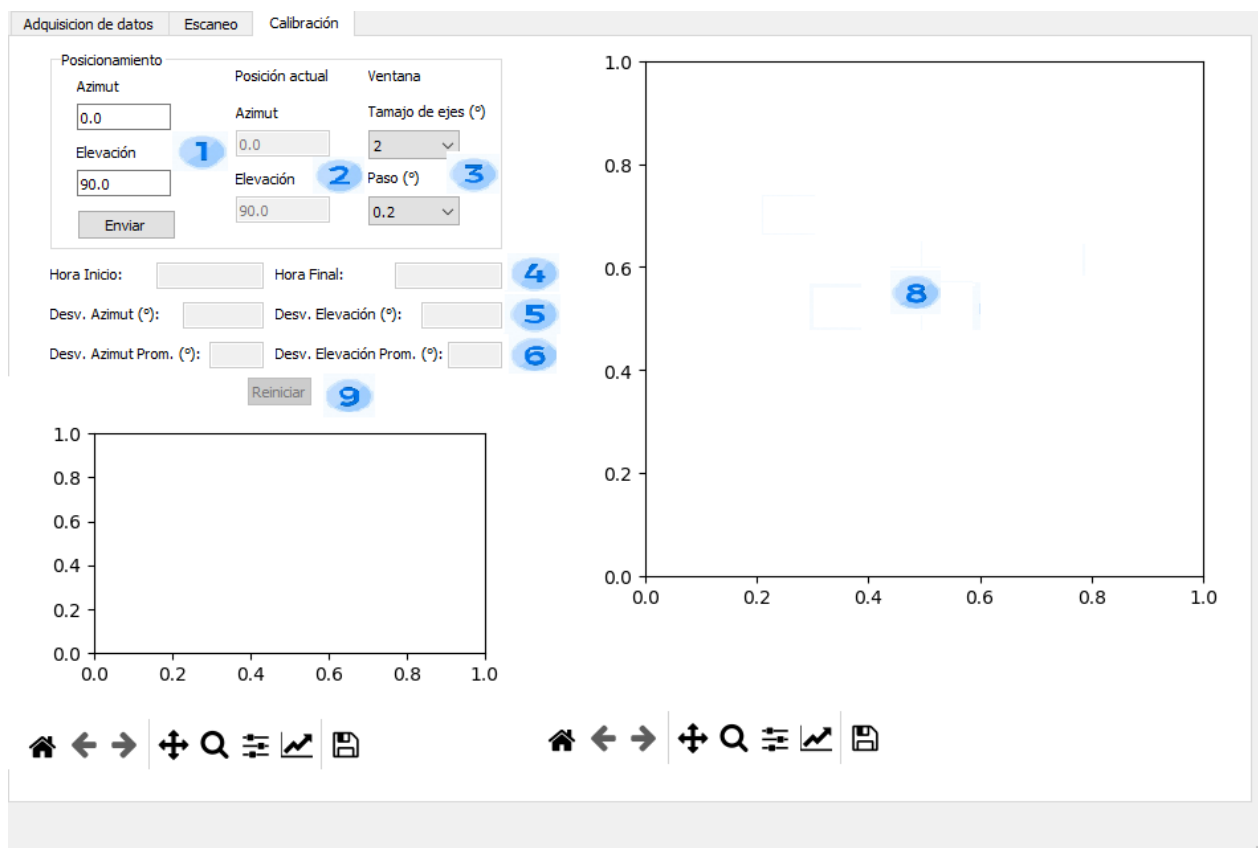


Figura 4.4: Página: Calibración.

- 1 Ingreso de la coordenada central de la matriz.
- 2 Visualización de la posición actual del radiotelescopio.
- 3 Definición del ancho de la matriz y la resolución de esta.
- 4 Visualización de la hora de inicio y finalización de la adquisición.
- 5 Visualización de la desviación encontrada durante la adquisición.
- 6 Visualización de la desviación promedio para las sesiones de adquisición realizadas.
- 7 Visualización gráfica de las desviaciones calculadas durante cada sesión.
- 8 Visualización gráfica de la zona del cielo observada.
- 9 Limpiar las desviaciones acumuladas.

4.2 Algoritmos

4.2.1 Ventana Adquisición de datos

El algoritmo que define el funcionamiento de la página se enfoca entonces en optimizar las sesiones de adquisición, definiendo un orden para la toma de datos. Una vez inicializada la ventana, se habilita el ingreso de la coordenada a la que deseamos enviar el radiotelescopio en la caja “Posicionamiento”. En esta se habilitan las cajas de texto para el ingreso del acimut y altitud donde queremos apuntar la antena, durante cualquier momento de la adquisición. Una vez es oprimido el botón “Enviar”, se ejecuta la función *SendPosition()* para las coordenadas ingresadas. Al completar el movimiento, se actualiza la posición como la coordenada final de la antena utilizando la función *Position()*. Por otro lado, al iniciar encontraremos desbloqueado el botón “Cielo de Fondo”, el cual cumple la función de captar la respuesta del radiorreceptor durante cinco segundos utilizando la función *ADQ()*. Por último, se construye la gráfica bidimensional de los datos tomados y se muestra el valor en voltios como se ilustra en el diagrama de flujo de la Figura 4.5.

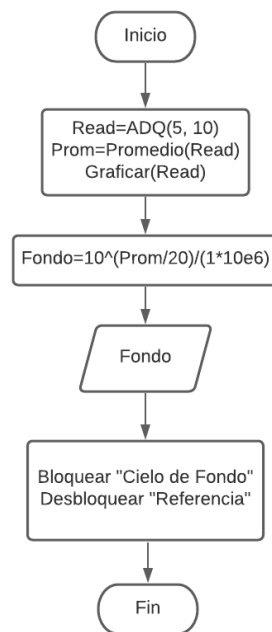


Figura 4.5: Diagrama de flujo al oprimir el botón “Cielo de fondo”.

Una vez finalizado el proceso se desbloquea el botón “Referencia”. Este cumple la misma función del anterior, pero en este caso se debe realizar la adquisición de datos al apuntar un objeto cuya temperatura sea conocida (ver Figura 4.6).

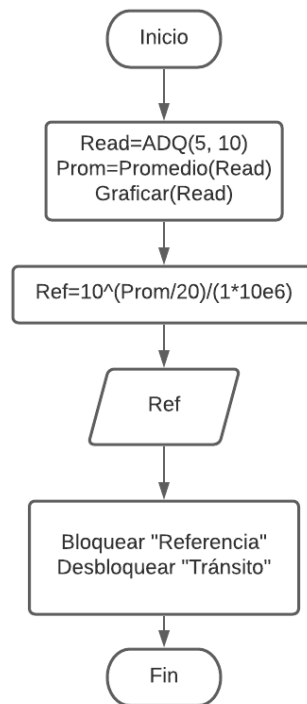


Figura 4.6: Diagrama de flujo al oprimir el botón “Referencia”.

Finalmente se habilitará el botón “Tránsito”. Antes de comenzar la captura del tránsito es necesario ingresar los siguientes parámetros.

- **Dec:** Declinación del objeto observado.
- **Time:** Tiempo de adquisición.
- **Tc:** Temperatura del objeto de referencia.
- **Ancho:** Ancho angular del objeto observado.

Al oprimir el botón, se ejecuta el algoritmo de la Figura 4.7, realizando adquisición de datos durante el tiempo ingresado. Una vez finalizada la sesión de adquisición, se guarda la información obtenida en un archivo excel. En este punto se utiliza la información adquirida en los tres momentos, con la cual se calcula la temperatura del sistema (T_{sys}) y la temperatura observada por la antena (T_A). Para complementar estos resultados se extrae el ancho de haz a partir del patrón de radiación (HBPWC) obtenido en el tránsito, el cual deriva en el área iluminada y eficiencia del plato reflector. Finalmente, se utilizan estos valores para determinar la temperatura de brillo del objeto y así terminar la sesión de adquisición.

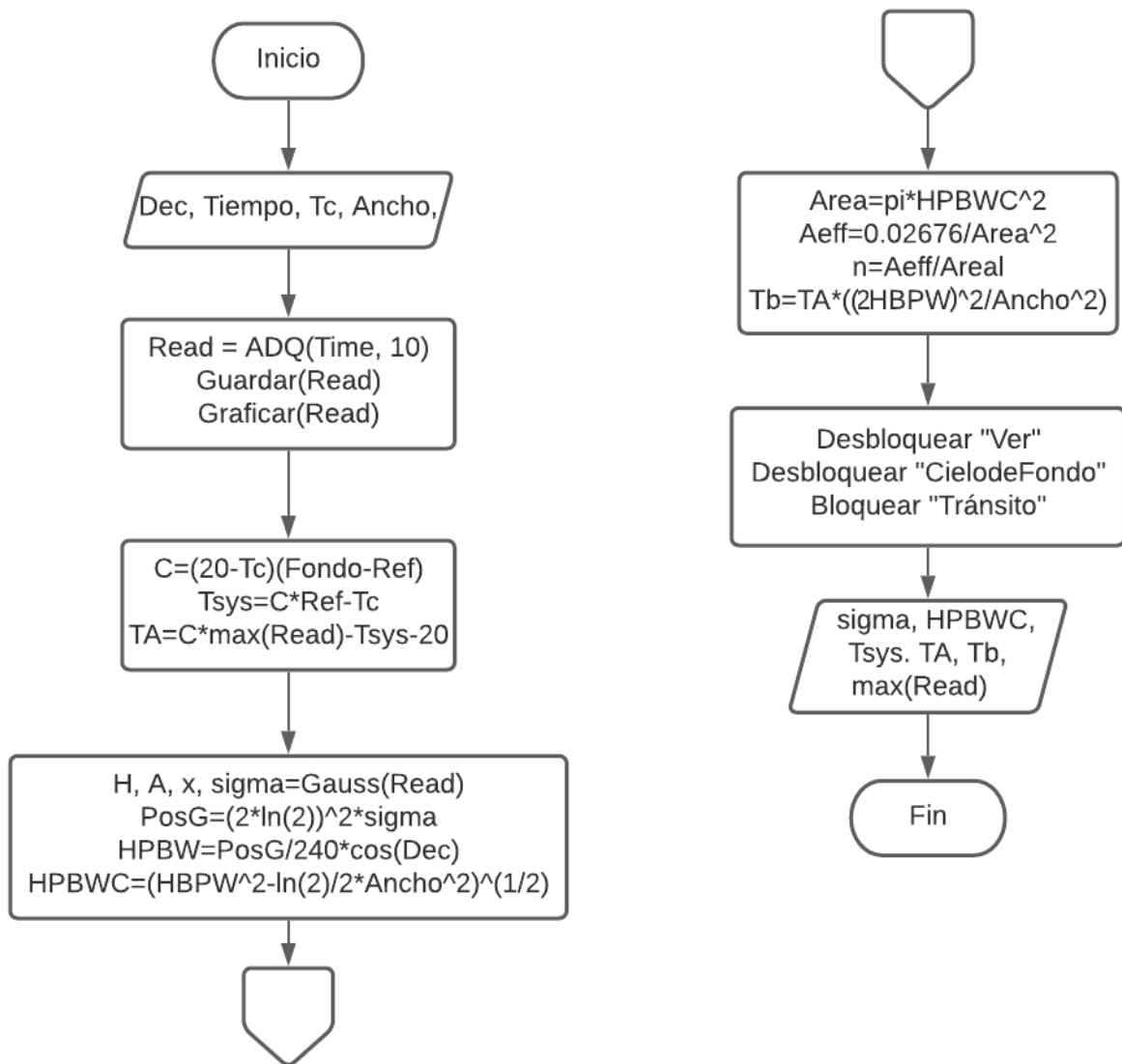


Figura 4.7: Diagrama de flujo al oprimir el botón “Tránsito”.

Para verificar la fidelidad de la aproximación Gaussiana con los datos obtenidos, se habilita el botón “Ver”, el cual permite visualizar el ajuste Gaussiano obtenido al utilizar la función *Gauss()* e implementado en el cálculo del ancho de haz de la antena. Al oprimir el botón se ejecuta el algoritmo mostrado en el diagrama de flujo de la Figura 4.8.

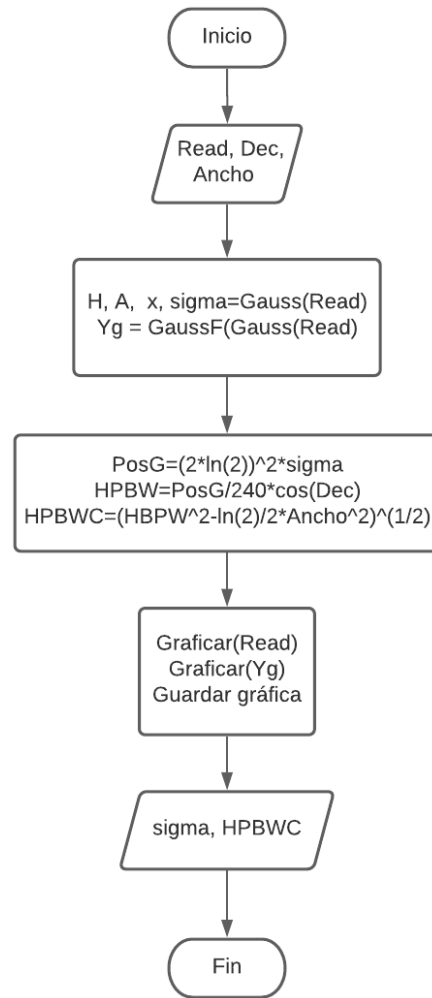


Figura 4.8: Diagrama de flujo al oprimir el botón “Ver”.

4.2.2 Ventana Escaneo

Ya que el algoritmo de la ventana se encarga de generar una imagen de una zona del cielo al presionar el botón “Enviar”, las coordenadas ingresadas serán el punto central de la misma. Para realizar una sesión de adquisición óptima es necesario ingresar los parámetros que definen el ancho y alto de la imagen, además de la coordenada central a observar. Antes de iniciar se deben ingresar los siguientes valores.

- **Picsize:** Altura de la imagen en pixeles.
- **Sample:** Frecuencia de muestreo durante cada movimiento.
- **Ancho:** Ancho angular de la ventana.
- **Time:** Tiempo tomado por la antena para realizar el movimiento.

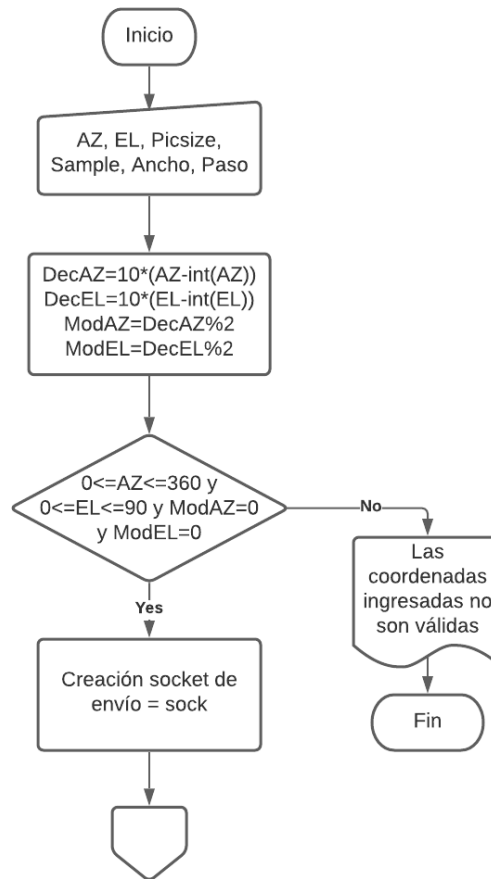


Figura 4.9: Ventana Escaneo: Primera parte.

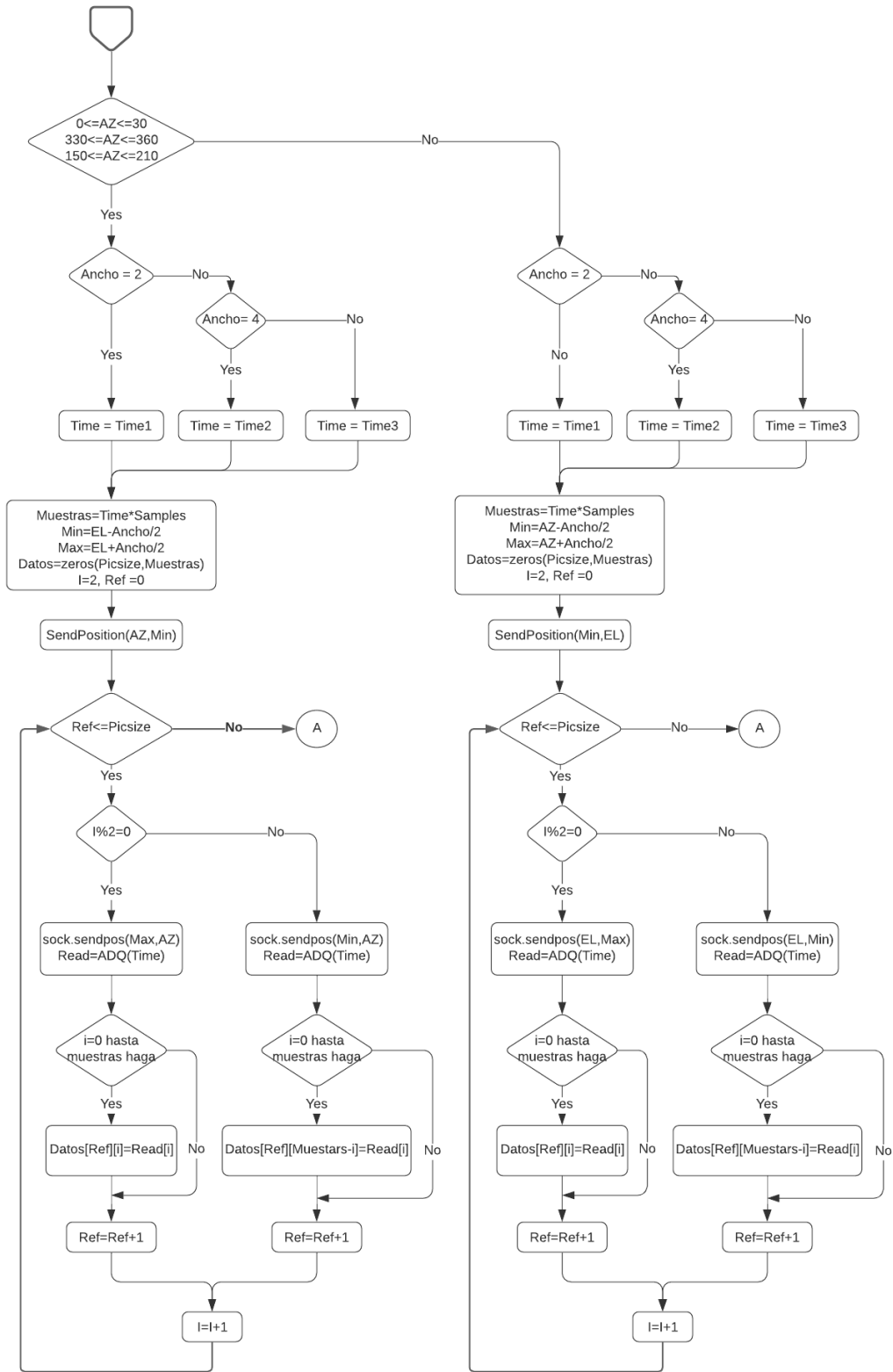


Figura 4.10: Ventana Escaneo: Segunda parte.

Dado que este proceso implica dejar una de las coordenadas constante mientras la otra realiza un movimiento cíclico de barrido, esperando captar el objeto cuando este pase por el haz de la antena, la elección de la coordenada que se mantendrá constante dependerá de la coordenada de acimut en la que se ubicará el radiotelescopio. De esta forma, aquellas coordenadas con un acimut en los rangos de $0^\circ - 30^\circ$, $150^\circ - 210^\circ$, $330^\circ - 360^\circ$, realizarán la adquisición manteniendo el acimut constante, mientras que, aquellas que no cumplan esta condición, realizarán la adquisición manteniendo la altitud constante. El proceso se puede realizar con un ancho de ejes de 2° , 4° y 6° durante un tiempo de adquisición sincrónico con el movimiento. El envío de las posiciones deseadas se logra a través de la función *SendPosition()*. Teniendo en cuenta que, el tiempo de adquisición debe ser sincrónico con el tiempo del movimiento, es necesario ingresar el tiempo medido para el movimiento de la antena en cada dirección y magnitud. Esta sección del algoritmo general se muestra en las Figuras 4.9 y 4.10. Para la lectura de datos se implementa la función *ADQ()*, teniendo como variable de entrada el tiempo escogido por el usuario para el movimiento.

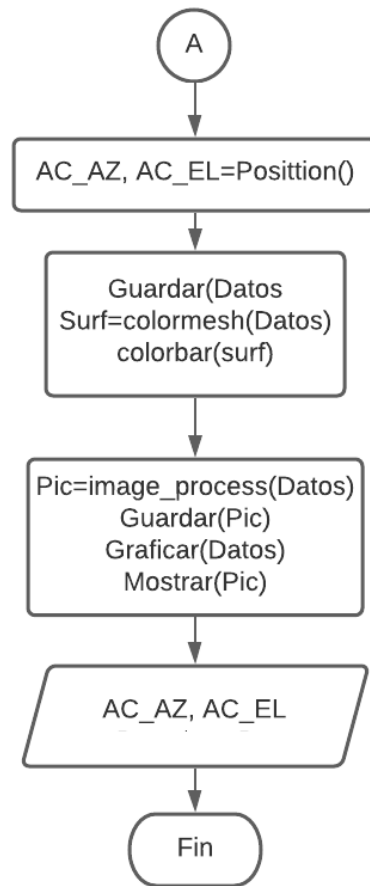


Figura 4.11: Ventana Escaneo: Tercera parte.

Una vez finalizada la adquisición, se definen, en el algoritmo de la Figura 4.11, los límites de los ejes bidimensionales de la gráfica, a partir de las posiciones de la antena y el tiempo de cada movimiento, para luego ser mostrada con un mapeo de color. Para la construcción del mapa bidimensional, que permite contrastar las intensidades captadas, se hace uso del módulo *Pcolormesh()*, extraído de la librería *Matplotlib*¹. Se establece entonces el eje horizontal como la coordenada en variación y el eje vertical a partir del momento en que se realizó el movimiento. Además, utilizando el módulo *Colorbar()*, se construye una barra de colores en representación al valor en intensidad de cada una de las posiciones para la figura gráfica. Por otro lado, para realizar la construcción de una imagen a partir de datos almacenados por las sesiones de adquisición, se utilizó el módulo *Image*, de la librería *Pillow*². Este permite

¹<https://matplotlib.org/>

²<https://pillow.readthedocs.io/en/stable/>

generar una imagen en escala de grises a partir de los datos normalizados, además facilita su almacenamiento y lectura.

4.2.3 Ventana Calibración

Esta ventana de la aplicación permite realizar la observación del cielo a través de una matriz cuadrada de tamaño predefinible al presionar el botón “Enviar”. En esta, cada punto representa un valor de intensidad medido para la posición en la que se encuentra apuntado el radiotelescopio. Como variable de ingreso se tomará el ancho de la ventana (Ancho), la resolución que esta tendrá (Paso) y la posición donde se enviará el radiotelescopio. Una vez se han ingresado el tamaño de la ventana y los pasos con que se ejecutará esta, se definen los puntos coordinados mínimo y máximo, enviando la antena al punto central donde se adquiere el primer valor número de la matriz como se muestra en la Figura 4.12.

Una vez se ha definido el tamaño de la matriz se pasa a generar una posición para cada elemento de esta. Así, cada vez que el telescopio cambia de posición, es guardado en el elemento consecuente de la matriz el valor de la intensidad medido. Por otro lado, el recorrido espacial a realizar dependerá de la posición ingresada. De esta forma se realiza la adquisición en una dirección de altitud descendente cuando el acimut ingresado es menor a 180° , mientras que las coordenadas mayores a este valor son correspondientes a una adquisición en altitud ascendente. En ambos casos se toma la coordenada central como punto de inicio. Como se muestra en las Figura 4.13, cuando la condición se cumple, el radiotelescopio se envía a la posición máxima de acimut y elevación, iniciando el recorrido de la matriz definida por el usuario, saltando la adquisición en la coordenada central, hasta llegar a la posición mínima de altitud de la matriz ingresada. Por otro lado, cuando la condición no es cumplida, se realiza el protocolo ilustrado en la Figura 4.14, iniciando el recorrido en la posición mínima de acimut y altitud, saltando la adquisición en la coordenada central y culminando en el punto máximo de altitud.

Una vez finalizada la sesión de adquisición se guardan los datos y se definen los límites de los ejes bidimensionales de la gráfica resultante, a partir de las posiciones máxima y mínima de la antena. Se construye la malla bidimensional para ser mostrada en un mapeo de color utilizando nuevamente las funciones *Pcolormesh()* y *Colrobar()*. Finalmente, se extrae la desviación entre el punto de mayor detección y la posición central de la ventana ejecutada, almacenando este valor constantemente para llevar así su valor promedio. De forma complementaria, se ilustra la desviación en cada ventana de observación efectuada en una gráfica bidimensional utilizando la función *Scatter()* de la librería *Matplotlib*. El algoritmo que define el protocolo final se ilustra en la Figura 4.15.

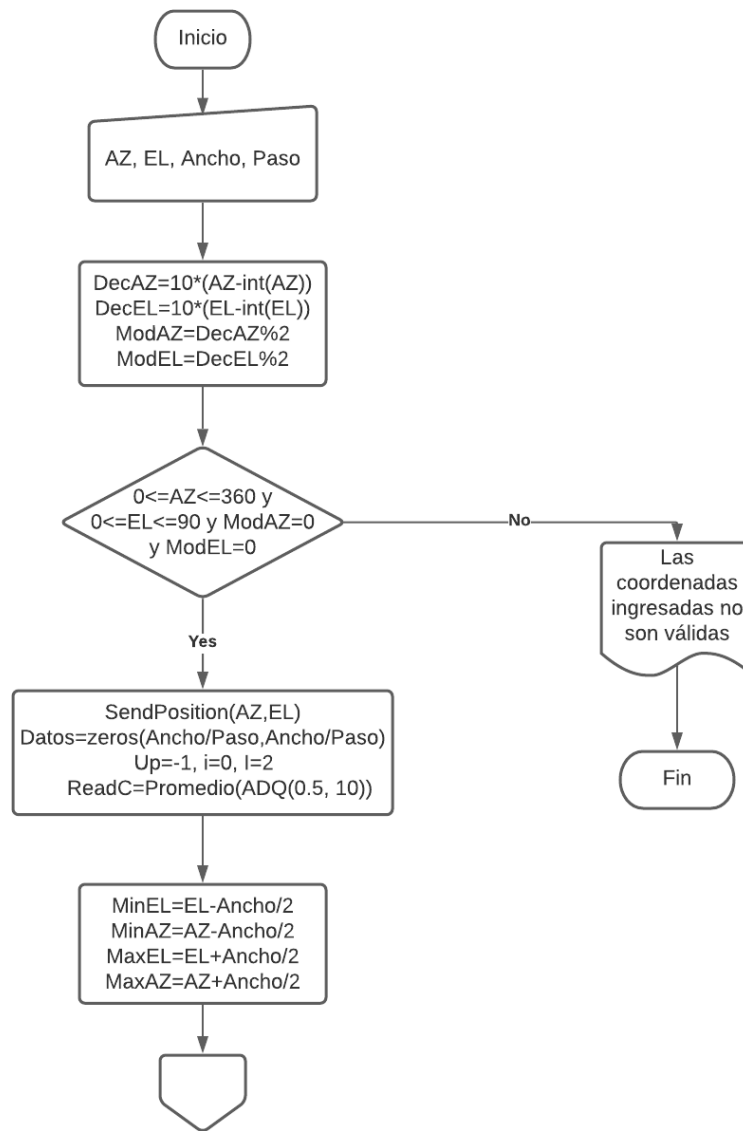


Figura 4.12: Ventana Calibración: Primera parte.

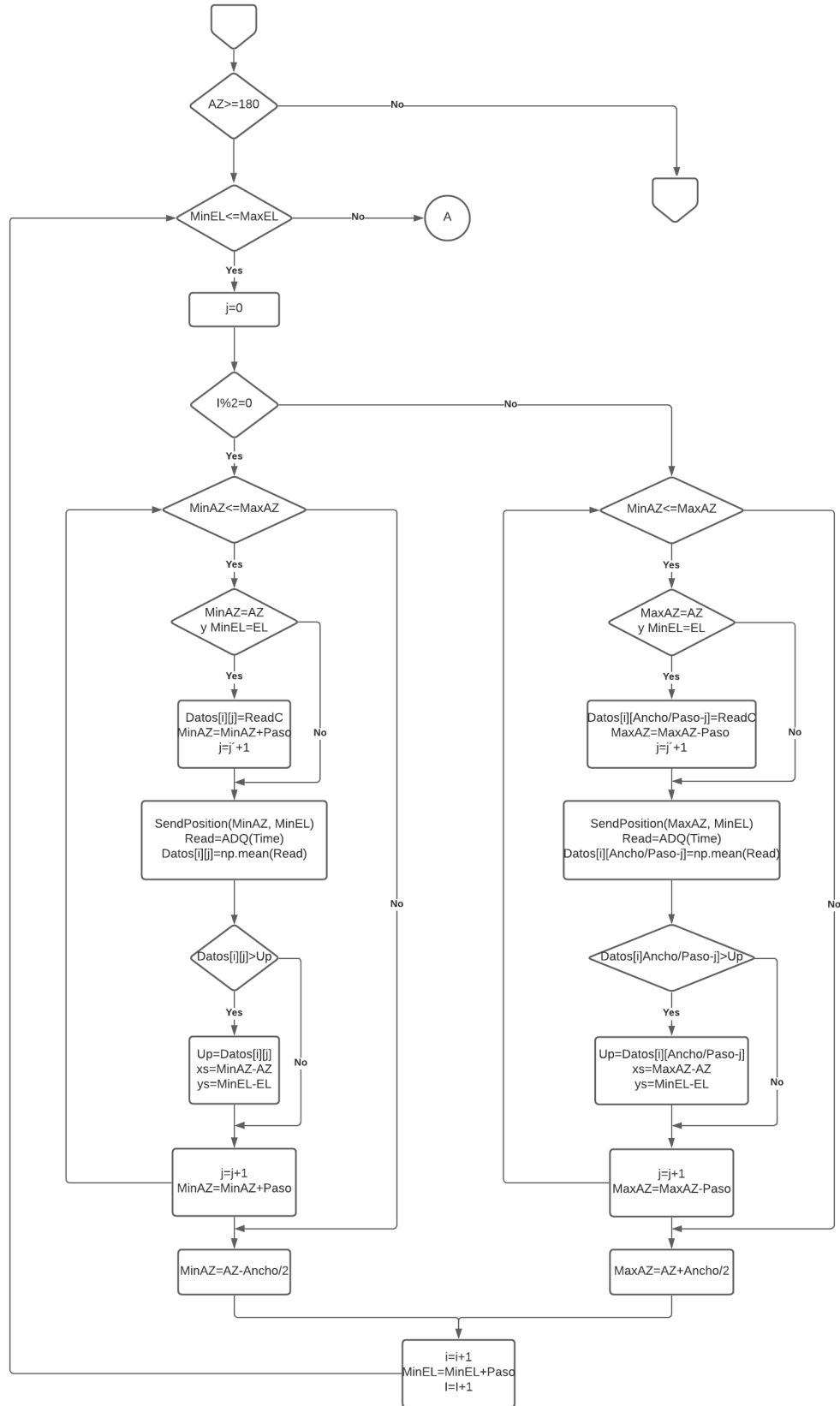


Figura 4.13: Ventana Calibración: Segunda parte.

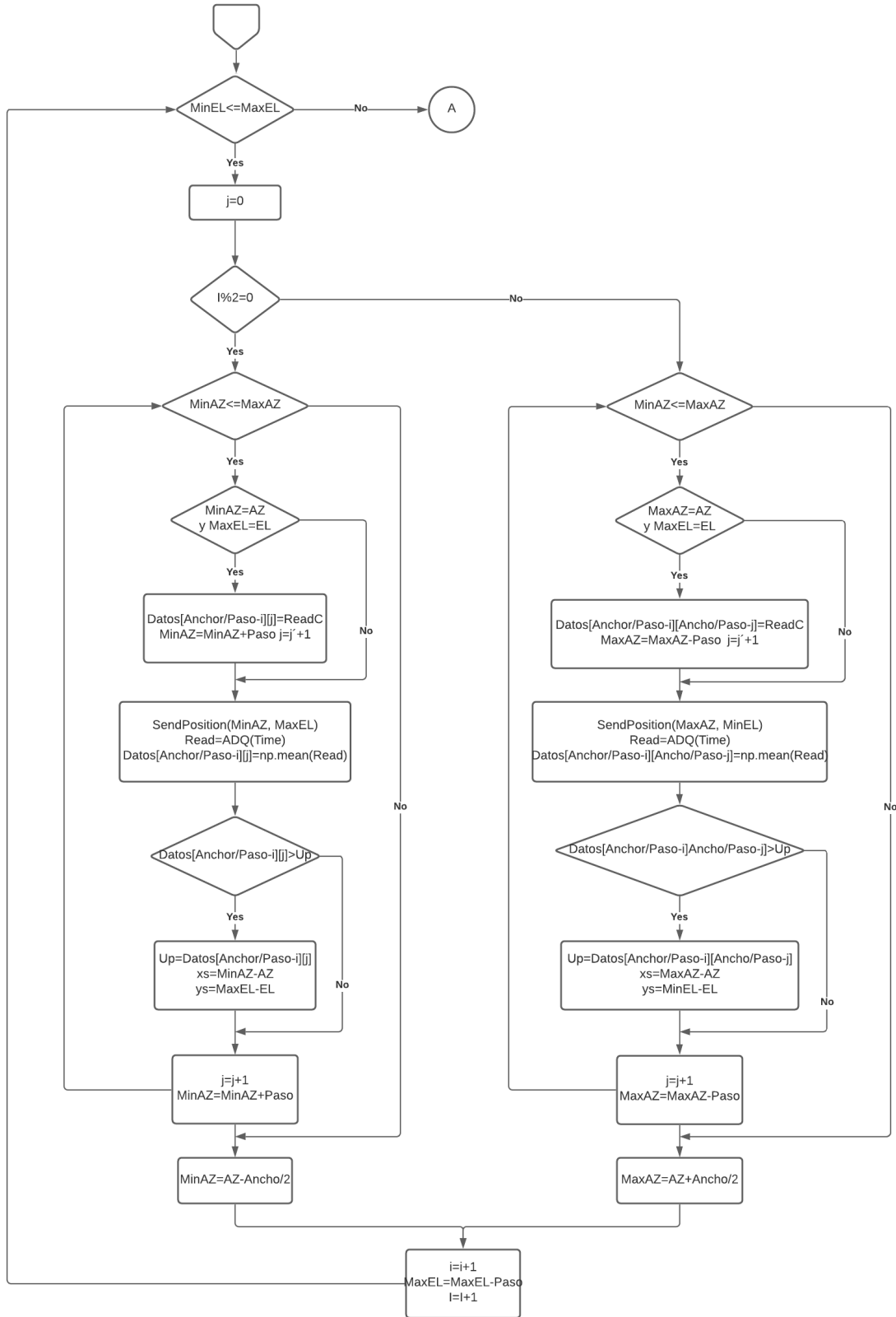


Figura 4.14: Ventana Calibración: Tercera parte.

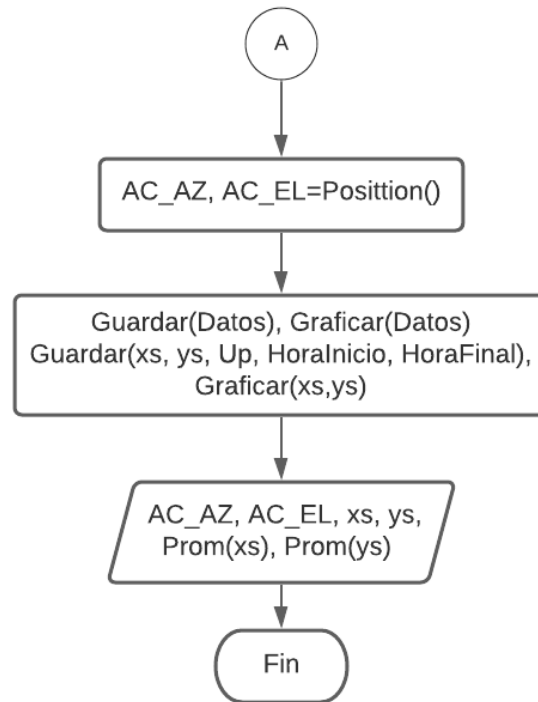


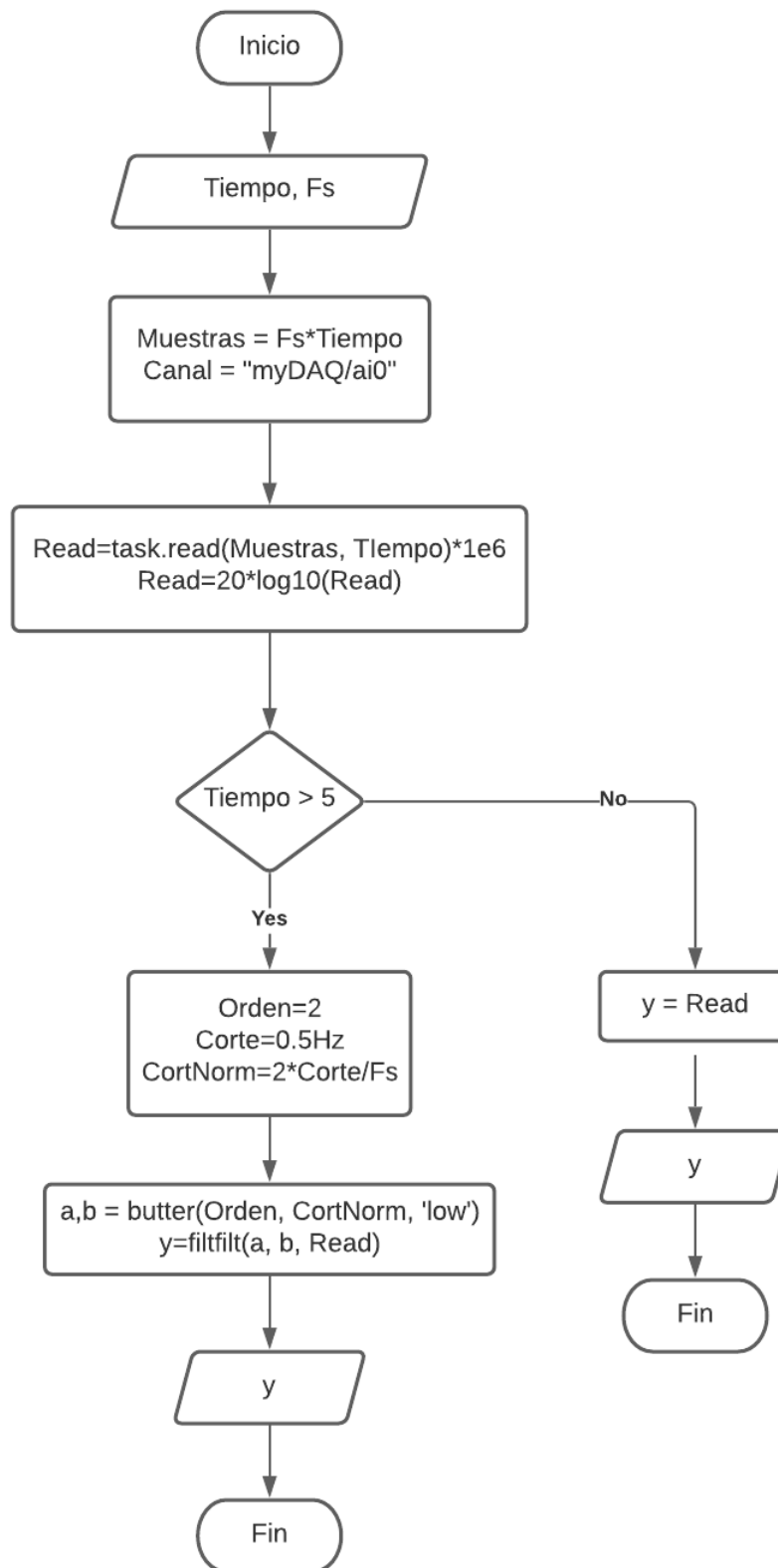
Figura 4.15: Ventana Calibración: Cuarta parte.

4.2.4 Comunicación DAQ

Como parte inicial de la construcción de la interfaz se realizó la función encargada de captar y procesar el voltaje recibido por medio de la tarjeta de adquisición de datos. Para esto se implementó la librería *NIDAQmx*³, la cual permite configurar el tiempo de adquisición y la frecuencia de muestreo fácilmente. Utilizando la función *Task.read()*, se logró la adquisición de datos durante los parámetros configurados. Una vez almacenada la información, esta es escalada, pasando su magnitud de V a $dBuV$, para luego ser procesada a través de un filtro Butterworth pasa bajos digital de segundo orden, con una frecuencia de corte de 0.5 Hz. Este proceso se logra al implementar las funciones *Butter()* y *filtfilt()* de la librería *Scipy*⁴. Cabe resaltar que el filtrado solo se realiza para señales con una duración mayor a 5 s. El algoritmo completo se almacenará en la función *ADQ()*, cuyo diagrama de flujo es ilustrado en la Figura 4.16. Esta tendrá como variable de entrada el tiempo de adquisición, dando en su salida el arreglo completo de los datos procesados.

³<https://nidaqmx-python.readthedocs.io/en/latest/>

⁴<https://www.scipy.org/>

**Figura 4.16:** Diagrama de flujo de la función *ADQ()*.

4.2.5 Envío y recepción de coordenadas

Antes de la realización de esta tesis, para realizar el movimiento en acimut y altitud de forma remota era necesario ingresar las coordenadas a través del software proporcionado por el fabricante para la montura. Ya que este está diseñado específicamente para el software de control instalado en la montura, la comunicación con otros programas se encuentra limitada. En solución a esto se realizó la adquisición del programa Pstrotator (Figura 4.17), diseñado para múltiples monturas con movimiento en acimut y altitud. La principal ventaja de este programa es la capacidad de comunicarse a través de protocolos de datagramas de usuario (UDP). Una vez activada la comunicación UDP es posible enviar y recibir coordenadas a la montura a través de programas externos.

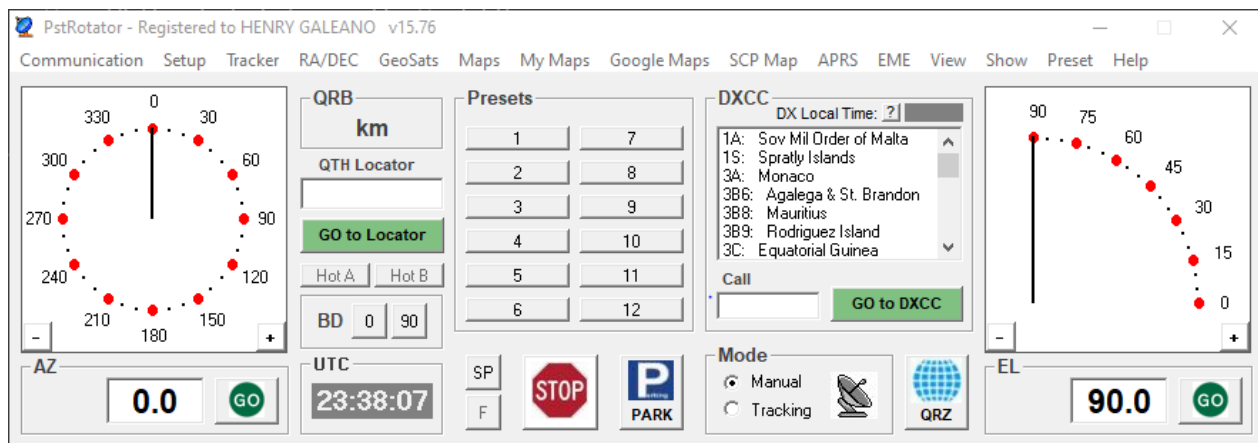


Figura 4.17: Ventana principal Pstrotator.

Teniendo habilitada la comunicación UDP con Pstrotator, se construyeron las funciones en Python para el envío y recepción de datagramas. Como primera instancia se realizó la función encargada de recibir y reportar la posición de la montura como se muestra en el diagrama de flujo de la Figura 4.18. En ella se utiliza la librería *Socket*⁵, agilizando la comunicación a través del sistema UDP. Se inicia entonces con la creación de los sockets para el envío y recepción de coordenadas con los puertos UDP 12000 y 12001 respectivamente. El primer socket se encarga de enviar el comando para solicitar el reporte de la posición actual al canal 12000, mientras que, el segundo escucha la respuesta en el canal 12001. Esta combinación permite realizar la adquisición de las coordenadas del radiotelescopio siempre que se utiliza la función *Position()*.

⁵<https://docs.python.org/3/library/socket.html>

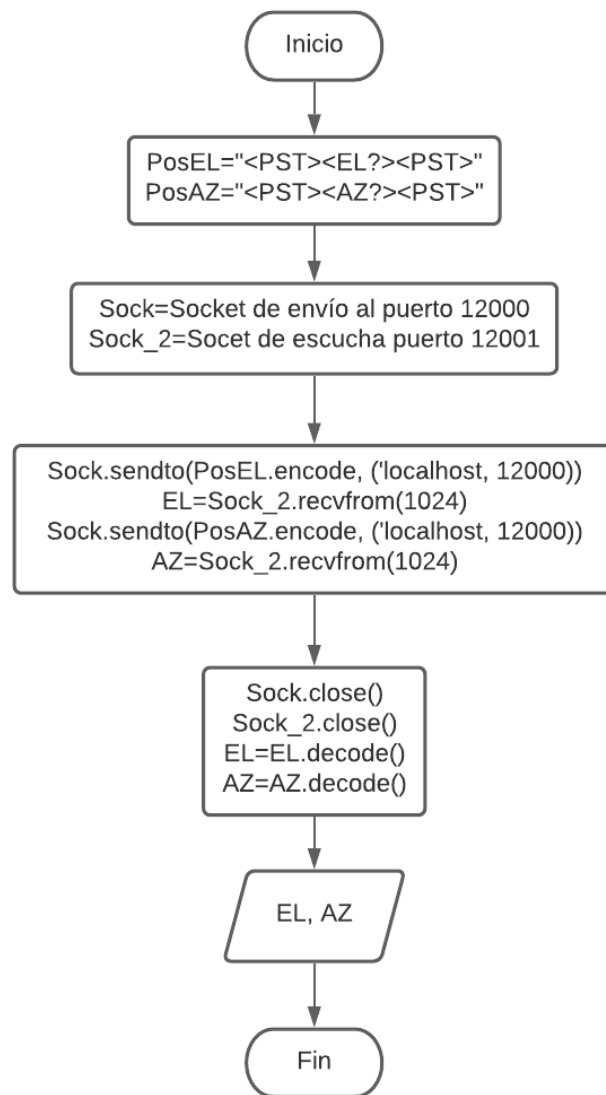


Figura 4.18: Diagrama de flujo función *Position()*.

Ahora, para el envío de coordenadas se construyó la función *SendPosition()*, mostrada en las Figuras 4.19 y 4.20. Esta toma los valores de acimut y altitud ingresadas por el usuario, validando que se encuentren bajo el rango permitido y así realizar el proceso de envío. Nuevamente se crean los sockets de envío y recepción, conectados a los puertos 12000 y 12001. El primer socket se encargará de enviar la coordenada deseada a PstRotator, para luego escuchar el puerto de actualización constantemente con el segundo socket, deteniendo el proceso cuando la montura haya alcanzado la posición deseada. Una vez culminado el algoritmo de la función, esta da como resultado la posición final del radiotelescopio.

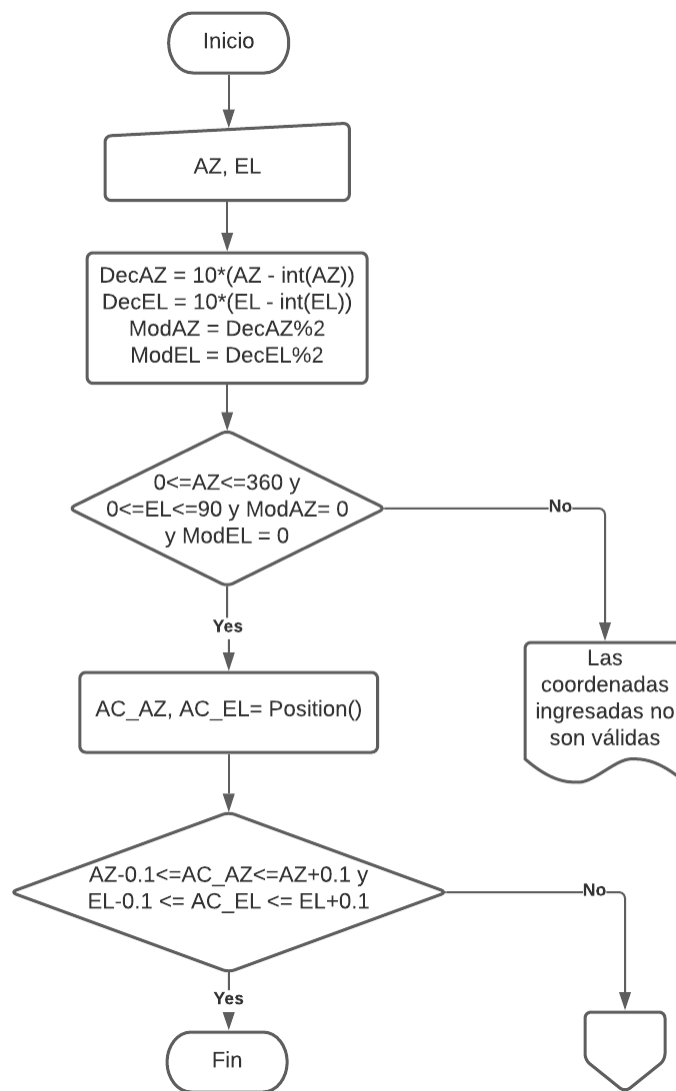
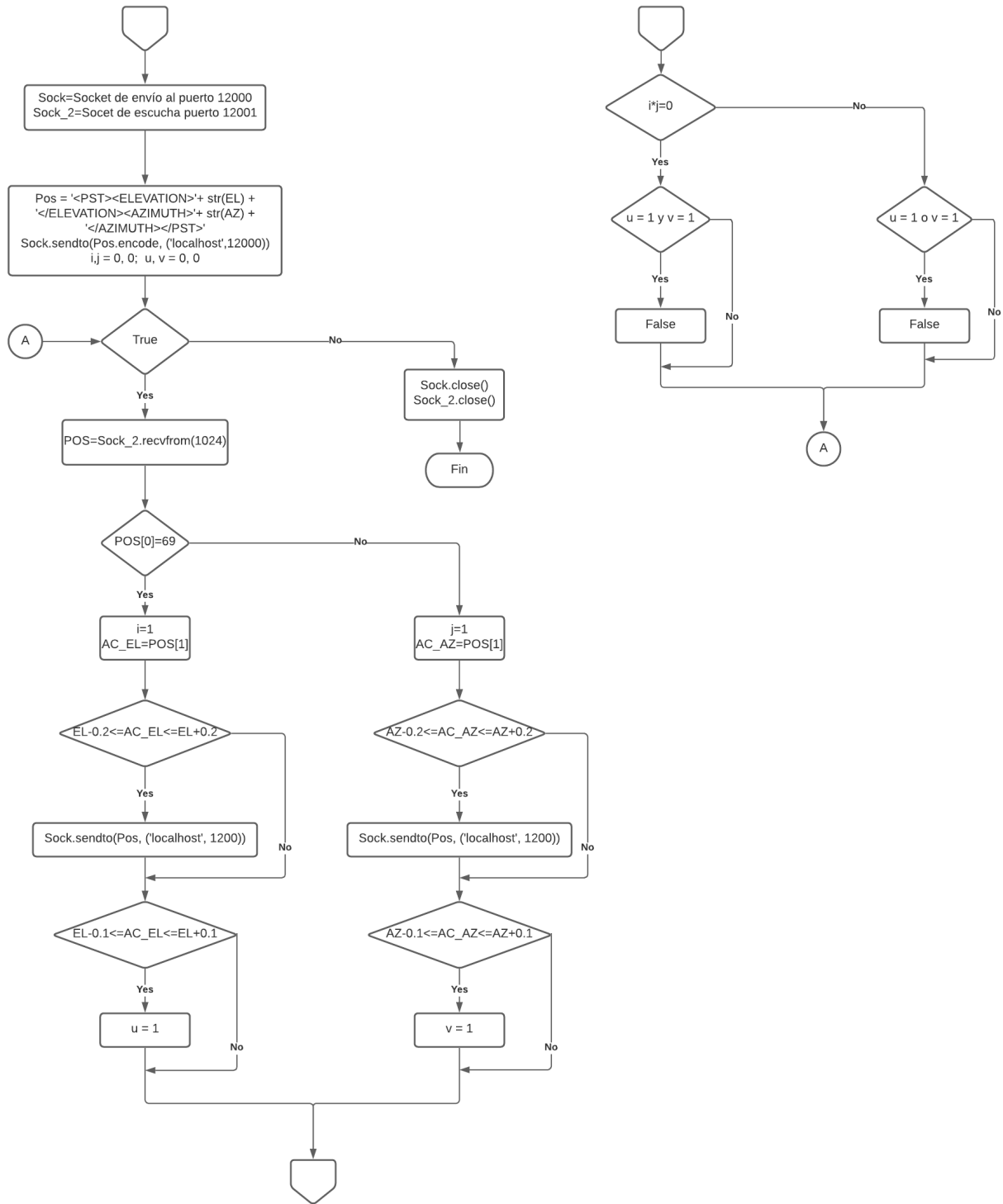


Figura 4.19: Función *SendPosition()*: Primera parte.

Figura 4.20: Función *SendPosition()*: Segunda parte.

4.2.6 Ajuste Gaussiano

Teniendo en cuenta que el cálculo del ancho medio de haz de la antena implica la interpretación de la curva gaussiana resultante de un tránsito solar, se construyó la función *Gauss()*. Como se observa en el diagrama de flujo de la Figura 4.21, esta entrega como resultado los parámetros de la función gaussiana más cercana a los datos ingresados. En la construcción del ajuste Gaussiano de un arreglo de datos, se implementó la función *Optimize()* de la librería Scipy. Antes de ser utilizada es necesario hallar el valor promedio de los valores a optimizar, además de su varianza. Posteriormente se ingresan estos valores a la función *Optimize()*, junto con la función esperada, en nuestro caso una campana de Gauss definida como $GaussF(x, H, A, x0, sigma) = H + Ae^{\frac{(x-x0)^2}{sigma^2}}$. Esta función se llamará *GaussF()*. Finalmente, se obtiene como resultado la interpolación más cercana a los datos ingresados con la función *optimize.curve_fit()*, la cual entrega las constantes *H*, *A*, *x0*, *sigma* optimizadas para el arreglo ingresado.

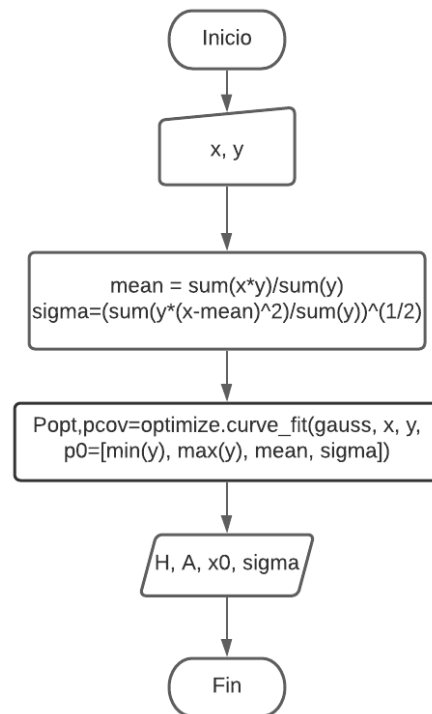


Figura 4.21: Diagrama de flujo *Gauss()*.

Capítulo 5

Potencial científico del instrumento

Dado que en la bibliografía se evidencia la observación de objetos como Saturno, Cassiopeia A y el centro galáctico con montajes similares, en esta tesis determinamos la posibilidad de observar estos cuerpos celestes con la instrumentación disponible. Además, se exploró la observación de otros objetos cuyo flujo de radiación sea detectable a 11.2 GHz. Una vez recopiladas las posibles radiofuentes seleccionamos aquellas enumeradas en la Tabla 5.1.

Objeto	Tipo
Cassiopeia A	Remanente de supernova
Cygnus A	Remanente de supernova
Nebulosa del Cangrejo	Remanente de supernova
Nebulosa de Orión	Nebulosa difusa
Saturno	Planeta
Centro de la Vía Láctea	Brazo galáctico
Sol	Estrella
Luna	Satélite natural

Tabla 5.1: Muestra de radiofuentes a observar.

Haciendo uso de la Nasa Extra Galactic Data Base, fue posible interpolar el flujo de los objetos de espacio profundo, mientras que el reporte diario The Learmonth Observatory se utilizó para determinar el flujo solar. Los valores obtenidos en la intensidad de flujo a 11 GHz para las muestras se enumeran en la Tabla 5.2.

Objeto	Flujo (Jy)
Cassiopeia A	423.69
Cygnus A	135.6
Nebulosa del Cangrejo	506.104
Nebulosa de Orión	309,74
Luna	56823.8
Sol	3749410

Tabla 5.2: Flujo de las radiofuentes a observar.

5.1 Ventanas de observación

El primer método que implementamos para la observación de objetos celestes es la de ventanas de observación. Esta consiste en escanear la zona donde se ubica el cuerpo en cuestión, con el objetivo de lograr una detección positiva. Este método se ejecutó realizando las técnicas de observación matricial y de escaneo, donde se busca obtener una zona bidimensional de la bóveda celeste. Ambas técnicas se describen a continuación.

5.1.1 Técnica matricial

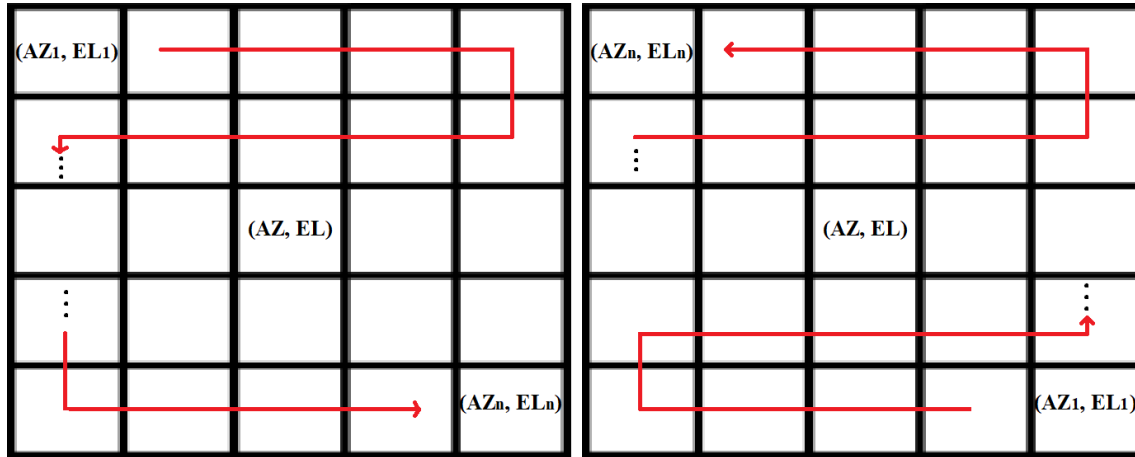
Al implementar la técnica matricial, se busca observar una malla cuadrada de tamaño $N \times N$ centrada en el objeto de interés. Dado que se espera obtener una respuesta en señal por cada posición coordenada, se realiza una adquisición de datos durante 0.5 s en cada posición de la matriz espacial generada. Siendo un valor en el rango del tiempo de integración de la antena parabólica utilizada. De esta forma cada elemento de la matriz A resultante (ver Figura 5.1), representa un valor de intensidad de señal en una coordenada dada.

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

Figura 5.1: Matriz espacial de las intensidades percibidas.

Teniendo en cuenta que cada posición de la matriz simboliza una coordenada en acimut y altitud, donde debe ser orientada la antena, la forma en la que esta será recorrida está sujeta a la posición del objeto a observar. Es por esto que aquellas coordenadas cuyo acimut sea

inferior a 180° , las cuales corresponden a objetos ascendentes, realizarán la adquisición desde una altitud mayor a la seleccionada, ejecutando el recorrido en dirección descendente. Por otro lado, para coordenadas superiores a 180° , la adquisición se iniciará desde una Elevación inferior a la seleccionada, llevando a cabo el recorrido en dirección ascendente. El camino seguido en ambos casos se ilustra en la Figura 5.2.



(a) Recorrido para valores de acimut menores a 180. (b) Recorrido para valores de acimut mayores a 180.

Figura 5.2: Direcciones del movimiento.

5.1.2 Técnica de escaneo

Al efectuar esta técnica se busca realizar un escaneo de deriva en la zona del cielo donde se ubica el objeto a observar. El procedimiento se ejecuta manteniendo una de las coordenadas de acimut o altitud en un valor constante, mientras se mueve la otra de forma oscilatoria en un rango de grados predeterminado y un tiempo aproximado de 30 minutos. Una vez obtenida la visualización gráfica de la zona, se espera obtener una lectura positiva cuando el objeto coincida con el punto máximo del ancho de haz de la antena, mientras pasa por este a velocidad sideral.

Teniendo en cuenta que la declinación del objeto influye directamente en su recorrido alrededor del cielo, antes de realizar el escaneo de la zona se debe seleccionar correctamente la dirección que se mantendrá constante. Es por esto que para aquellos objetos cuya posición en acimut se encuentre entre $330^\circ - 30^\circ$ y $150^\circ - 210^\circ$, se debe realizar la adquisición dejando la coordenada acimut constante. Mientras que, para aquellos fuera de este rango, la adquisición deberá hacerse dejando altitud en un valor constante. Estos parámetros se ilustran en la Figura 5.3.

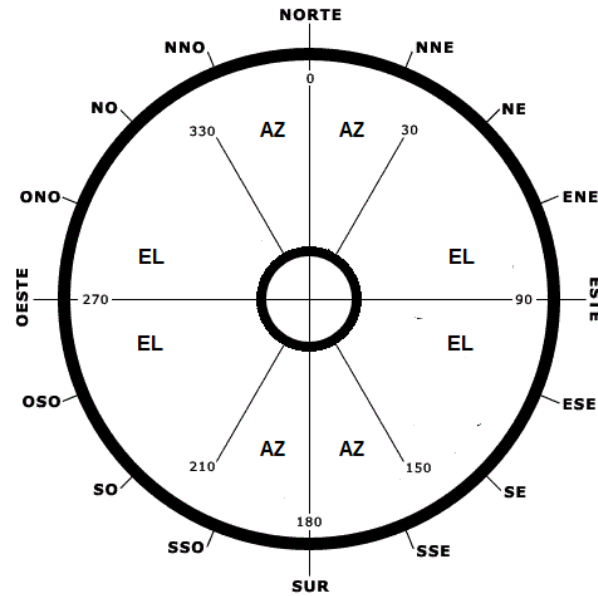


Figura 5.3: Coordenada constante bajo valores de acimut diferentes.

5.2 Tránsitos

El método de tránsito consiste en direccionar el radiotelescopio hacia una coordenada por la que el cuerpo celeste va a cruzar. Para lograr la observación completa del evento se comienza la toma de datos unos minutos antes de que el objeto comience a pasar en frente de la antena y se termina hasta que este salga del campo visual de la misma. El resultado del anterior procedimiento es la curva gaussiana representativa del patrón de radiación de la antena, tal como se muestra en la Figura 3.10.

5.3 Temperatura de brillo

La temperatura de brillo, también conocida como temperatura de radiación, es la temperatura de un cuerpo negro en equilibrio térmico con su entorno y no es una temperatura como se entiende normalmente. Este parámetro físico caracteriza la radiación, y dependiendo del mecanismo que la emite puede diferir considerablemente de la temperatura física de un cuerpo radiante. Partiendo de la radiación de cuerpo negro es posible determinar la temperatura de brillo de un objeto para una frecuencia específica. Para nuestro caso en particular, los paquetes de energía de la onda captada son mucho menores a la energía por radiación térmica, por lo que podemos aplicar la ley de Rayleigh-Jeans para determinar la radiancia espectral.

$$B_v = \left(\frac{2 \cdot k_b \cdot T}{\lambda^2} \right) \quad (5.3.1)$$

Donde

- B_v =Radiación espectral
- k_b =Constante de Boltzman
- T =Temperatura absoluta del objeto
- λ =Longitud de onda

En la ecuación (5.3.1) se visualiza la proporcionalidad entre la radiancia espectral de un cuerpo en equilibrio térmico a temperatura T y la frecuencia de radiación. A partir de esto es posible afirmar que, para la frecuencia seleccionada de 11.2 GHz la lectura de la antena (V_A) representa una medida de temperatura (T_A). Para obtener un valor de temperatura en función del voltaje recibido, el objeto observado debe cubrir en su totalidad el ancho de haz de la antena, lo cual arroja la relación de la ecuación (3.2.2).

$$T_A = C \cdot V_A - T_{cold} - T_{sys} \quad (5.3.2)$$

Obteniendo los valores de la constante C , la temperatura de fondo (T_{cold}) y del sistema (T_{sys}) a partir de lo mencionado en la sección 3.2.1, se facilita el cálculo de la temperatura medida por la antena. Ahora, se realiza la corrección de T_A teniendo en cuenta el ancho angular (θ) del objeto y el ancho angular del haz de la antena. Para esto, se determina su ángulo sólido (Ω), el cuál representa su área angular.

$$\Omega = 2\pi(1 - \cos\theta) \quad (5.3.3)$$

Puesto que ambos ángulos son inferiores a 10° , la ecuación puede reducirse a la siguiente forma.

$$\Omega = 2\pi\theta^2 \quad (5.3.4)$$

Finalmente, la relación entre la temperatura de brillo del objeto (T_b) y la temperatura medida por la antena, se encuentra ligada al ancho angular del FPBW de la antena (Ω_A) y del objeto (Ω_B), como se muestra en la ecuación (5.3.5), reducible a la ecuación (5.3.6)(Chanka, 2018).

$$T_b = \left(\frac{\Omega_A}{\Omega_B} \right) T_A \quad (5.3.5)$$

$$T_b = \left(\frac{FPBW^2}{\theta_{obj}^2} \right) T_A \quad (5.3.6)$$

La temperatura de brillo representa la temperatura a la que un cuerpo negro irradiaría una intensidad de radiación electromagnética idéntica a la del objeto observado para una frecuencia específica (Gulkis y de Pater, 2003). Esta temperatura se asemeja a la temperatura física si el objeto observado se comporta como un cuerpo negro (de Parter y S. Kurth, 2014). Por otro lado, para determinar la intensidad de flujo captada por la antena al apuntar una radiofuente se recurre a la ecuación (3.2.4). De esta se despeja el flujo S , una vez se ha determinado el cambio en respuesta ΔU_{dB} cuando el radiotelescopio apunta hacia el objeto en cuestión.

$$S = (10^{\Delta U_{dB}/10} - 1) \frac{kT_{sys}}{A_{eff}} \quad (5.3.7)$$

Capítulo 6

Estrategia didáctica para la enseñanza de la radioastronomía

Teniendo en cuenta el proceso realizado para la calibración del radiotelescopio y la evaluación de sus capacidades científicas, se estableció que otro potencial del instrumento se encuentra en la enseñanza de la radioastronomía. Dado que la variabilidad de las condiciones meteorológicas en la región afectan levemente las condiciones de observación para ondas de radio, la implementación de una práctica de laboratorio enfocada a la radioastronomía en banda KU representa una oportunidad de aprendizaje cuantitativo. En consecuencia, se desarrolló una metodología para la enseñanza de la radioastronomía dirigida a estudiantes de ingeniería, a través de una estrategia didáctica basada en el modelo socioconstructivista, con el objetivo de promover el desarrollo integral de los estudiantes al involucrar componentes teóricos y prácticos.

6.1 Parámetro físico a observar

Sabiendo que la mejor respuesta del instrumento se logra al realizar observación solar, se enfocará la estrategia didáctica en la determinación de la temperatura de brillo de nuestra estrella principal. Dado que la frecuencia de operación del radiotelescopio se centra en 11.2 GHz, la zona de observación se encuentra ubicada entre la corona solar baja y la zona de transición, como se muestra en la Figura 6.1, donde la temperatura de brillo puede alcanzar hasta los 15000 K. El procedimiento involucrado en el cálculo de dicho parámetro físico, implica que el estudiante interactúe con los instrumentos de adquisición. Además, a medida que este avanza en el proceso, se fomenta la comprensión de los conceptos básicos de la física que fundamentan la emisión de radioondas y el funcionamiento de los radiotelescopios.

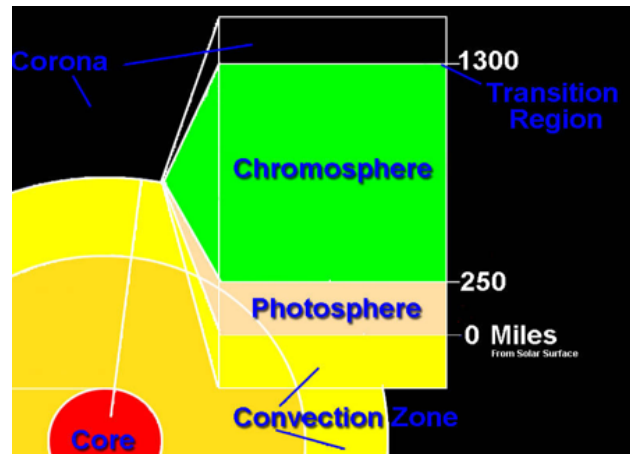


Figura 6.1: Ubicación de la capa solar a observar. Tomado de: https://www.nasa.gov/mission_pages/iris/images/index.html

6.2 Estrategia didáctica socio constructivista

Dados los resultados positivos arrojados en (García Ramírez, 2019), escogemos el desarrollo de la estrategia didáctica basados en la corriente socio constructivista. Por ende, se espera evaluar los saberes del participante antes y después de la intervención. Es necesario necesario dividir la experiencia en distintos puntos coyunturales que serán llamados Zona de Desarrollo Real (ZDR), Zona de Desarrollo Potencial (ZDP) y Zona de Desarrollo Próximo (ZDPr).

- **ZDR:** Siendo el primer momento de la práctica, se centrará en recolectar la información previa del participante. En este punto se espera evaluar los presaberes del estudiante, además de la comprensión que este tenga del tema a tratar.
- **ZDP:** En esta etapa se realizará la intervención implementando el instrumento para la enseñanza de radioastronomía. Aquí se suministrarán las herramientas necesarias para que el estudiante interactúe con el entorno que lo rodea y se familiarice con el funcionamiento de los instrumentos involucrados en la estrategia. Se espera entonces estimular el potencial del estudiante para que este tenga un aprendizaje a través de su autonomía.
- **ZDPr:** Finalmente, se evaluarán los conceptos asimilados una vez finalizada la intervención y así determinar el impacto causado en los estudiantes por el desarrollo de la experiencia.

6.3 Técnicas de recolección de la información

6.3.1 Observación participante

La observación participante implica que el monitor encargado de guiar la práctica de laboratorio recolecte toda la información involucrada en la ejecución de esta con el grupo participante. Aquí se describirá el tipo de actividad a realizar, los objetivos de esta y la pregunta esencial que se espera responder. Una vez se ha finalizado la práctica, el documento deberá reflejar si los objetivos se alcanzaron satisfactoriamente y si debe realizar algún cambio desde el punto de vista del monitor. El formato para la observación participante se muestra en el Apendice A.1.

6.3.2 Grupo de discusión

El grupo de discusión se realizará a través de una encuesta antes y después de la intervención. Se espera entonces la participación de todos los estudiantes, donde cada uno dará su opinión respecto a la pregunta realizada. Este elemento será determinante en los resultados arrojados por la ZDR y la ZDPr, ya que, las preguntas allí formuladas se verán enfocadas en conceptos básicos de la física que fundamentan la emisión de radioondas y el funcionamiento de los radiotelescopios. El formato para construido para el grupo de discusión se muestra en el Apendice A.2.

6.4 Instrumento para la enseñanza de la radioastronomía

El instrumento para la enseñanza de la radioastronomía es una guía de laboratorio que involucra instrumentos como una antena parabólica en malla de 1.2 m de diámetro con montura altacimutal motorizada, una interfaz gráfica de usuario desarrollada en Python y un radorreceptor de construcción propia. Durante la práctica, centrada en la observación del Sol, los estudiantes interactúan con los instrumentos y algunos de sus parámetros, como la relación entre el voltaje de respuesta y el cambio en la temperatura medida, además del ancho de haz para la observación en la banda KU (10-12 GHz). En búsqueda de dar un orden coherente al desarrollo de la guía se estructuró su contenido de la siguiente forma:

- **Resumen:** Describe brevemente el instrumento a operar, plantea la necesidad de definir algunos parámetros y metodologías necesarias para el desarrollo de la práctica, como también especifica el fenómeno a estudiar, que corresponde a la temperatura de brillo del Sol. Lo anterior con el propósito de dar una visión general de lo que se desarrollará en las siguientes secciones.

- **Objetivos:** Menciona lo que se desea obtener por parte de los estudiantes con el desarrollo de la práctica de laboratorio.
- **Fundamentos teóricos:** Plantea los conceptos y fundamentos necesarios para el desarrollo de la práctica y el entendimiento del fenómeno a estudiar, también menciona algunas de las características del instrumento involucradas en la práctica. Esta sección busca abarcar de forma breve y concisa las áreas de radioastronomía y radiación solar, antenas y radiotelescopios, temperatura de brillo, temperatura de la antena y diámetros angulares.
- **Materiales:** Describe cada una de las partes del radiotelescopio, el equipo involucrado para su manejo y la adquisición de datos.
- **Procedimiento:** Menciona las indicaciones iniciales para dar inicio con el el proceso de adquisición de datos, el cual se divide en 3 partes o zonas: toma de cielo de fondo, toma de una temperatura de referencia y toma de la iluminación del Sol. Finalmente describe los datos obtenidos y consideraciones que se deben tener en cuenta para el análisis de datos.
- **Análisis de datos:** Plantea 3 ejercicios que deben ser resueltos por los estudiantes con los datos obtenidos durante la práctica, para después buscar la interpretación de dichos resultados a través de preguntas de razonamiento crítico y lógico.

6.4.1 Desarrollo de la guía

El desarrollo de la práctica de laboratorio parte con la previa lectura de todo el contenido de la guía, esto con el propósito de tener una idea global de ¿Qué es lo que se va a desarrollar?, ¿cuál es el fenómeno que se va a investigar?, como también para tener presente los instrumentos que se manipularán durante la sesión. Una vez leída la guía se da inicio con la práctica desde la sección de procedimiento.

Tal como se menciona en la guía, se realizan las conexiones entre el PC, la DAQ, el Buscador de Satélites y la caja de alimentación y filtrado, como también se enciende la fuente SPID PS-02 y el controlador SPID MD-02. Por último se ejecutan los softwares PstRotator y CRT.

Siguiendo el procedimiento planteado en la guía, se pasa a realizar la primera de las tres tomas de datos, correspondiente a la toma de cielo de fondo. Para ello se direcciona la antena hacia el cenit (0° , 90°) y se pulsa el botón “Cielo de fondo”. Luego de 5 segundos se obtiene el resultado presentado en la Figura 6.2.

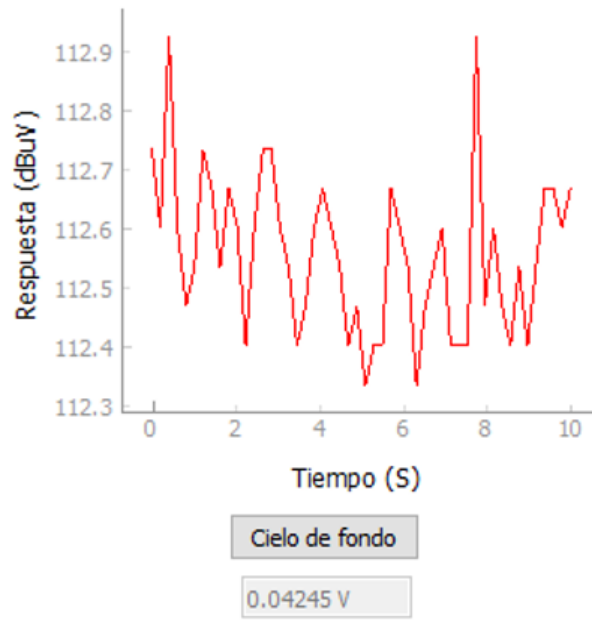


Figura 6.2: Toma de Cielo de fondo.

El valor en voltaje presente debajo de la gráfica corresponde a V_{sky} y se relaciona con T_{sky} que se asume en 20 K.

Una vez finalizada la primera toma de datos, se procede con la segunda que corresponde a la toma de una temperatura de referencia. Para ello se dirige la antena a las coordenadas (224° , 0°) de tal forma que quede direccionada hacia la cúpula del OAUTP. Una vez el instrumento llega a la posición se pulsa el botón “Referencia” y luego de 5 segundos se obtienen los resultados presentes en la Figura 6.3.

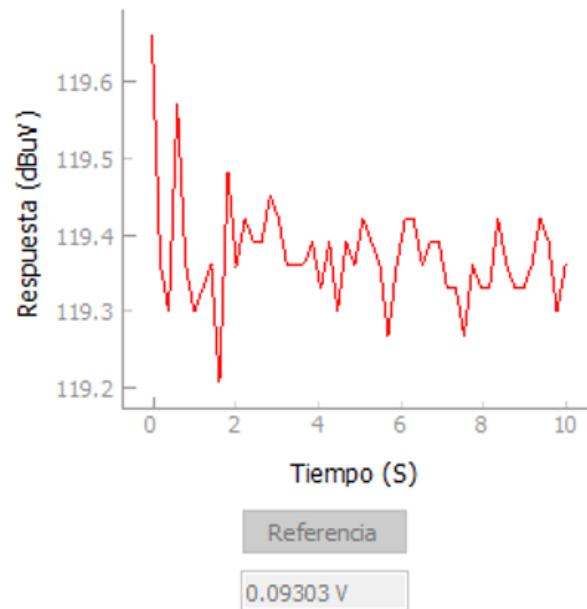


Figura 6.3: Toma de respuesta a una temperatura de referencia.

El voltaje resultante y visible debajo del botón “Referencia” corresponde a V_{ref} el cual se relaciona con la temperatura de la cúpula. Para obtener dicha temperatura se hace uso de un pirómetro o termómetro infrarrojo, con este se obtiene una temperatura T_{ref} de 27 °C que deben presentarse en Kelvin, así el valor de T_{ref} es de 300K.

Finalmente se procede con la última toma de datos que corresponde a la adquisición de un tránsito del Sol. Para ello se rellenan los cuadros de texto que se ubican en la zona inferior izquierda con la información que se solicita y se introduce la posición del Sol con un adelanto de 15 min. Hecho esto se presiona el botón “Tránsito”. Pasados 30 min se obtiene un tránsito del Sol, tal como se muestra en la Figura 6.4.

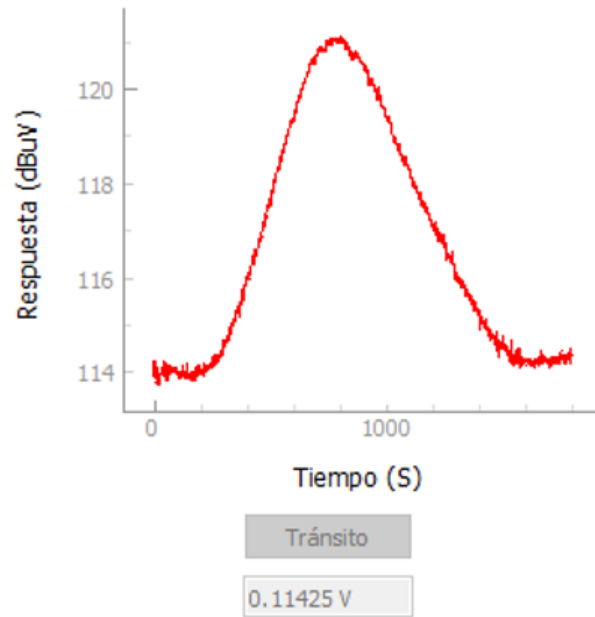


Figura 6.4: Toma del tránsito solar.

El voltaje presentado en el cuadro de texto debajo del botón “Tránsito” corresponde a V_a el cual se relaciona con T_a . Una vez finalizado el proceso de adquisición de datos, se da paso a la sección de procesamiento de datos. En este apartado se resuelven los 3 ejercicios planteados, tal como se muestra a continuación.

Cálculo del HPBW

Dado que el ancho de la curva del tránsito es obtenida en segundos, es necesario hacer su conversión a $^\circ$, obteniendo el siguiente resultado.

$$HPBW = 330.46s \frac{1^\circ}{240s} \cos(-12.7^\circ)$$

$$HPBW = 1.03^\circ$$

$$FPBW = 2.06^\circ$$

Cálculo del cociente entre el área angular del haz de la Antena y el área angular del objeto observado

El cálculo a realizar es el presentado en la ecuación 6.4.1. Aquí, Ω_a corresponde al área angular del haz de la antena, que puede ser calculado como el área de un círculo de radio HPBW, valor que se obtuvo en el primer ejercicio. Haciendo este cálculo se obtiene un valor de 13.4° para Ω_a . Por otra parte, Ω_b corresponde al área angular del objeto observado, valor

que puede ser calculado de la misma forma anterior pero con el radio angular del Sol. Hecho el cálculo se obtiene un área angular del Sol de 0.78° .

$$\frac{\Omega_a}{\Omega_b} \quad (6.4.1)$$

Obtenidos estos valores se calcula el cociente entre ambas áreas angulares, relación que obtuvo un valor de 16.97.

Cálculo de la temperatura de brillo del Sol

Para realizar este cálculo se parte de las ecuaciones 3.2.1 y 3.2.2, que además se presentan en la guía de laboratorio. Durante el proceso de adquisición se lograron obtener los valores para T_{sky} , V_{sky} , T_{ref} y V_{ref} . Con estos valores, y con las ecuaciones mencionadas, se resuelve el sistema de dos ecuaciones y dos incógnitas. Una vez resuelto se obtienen los valores de C y T_r , los cuales son 5617.83 K/V y 218.47 K respectivamente.

Hallados estos valores, es posible calcular la temperatura de brillo del Sol calculando primero T_a , que es la temperatura relacionada a la respuesta de la antena ante la radiación del Sol. Para ello se usa la ecuación 5.3.2, obteniendo un valor para T_a de 403.36 K. Ahora se procede a hacer el ajuste debido a la diferencia de área angular entre el haz de la antena y el Sol con la ecuación 5.3.6.

$$T_b = \frac{FPBW^2}{0.5^2} \cdot 403.36K$$

$$T_b = 6855K$$

Capítulo 7

Resultados

7.1 Capacidades científicas del radiotelescopio

7.1.1 Observación de radiofuentes de muestra

A partir de los resultados obtenidos en la calibración del sistema de apuntado y de recepción, pasamos a observar las radiofuentes elegidas como muestra. De esta forma se buscó verificar la correspondencia entre los resultados teóricos y prácticos. En consecuencia se ejecutó la observación de cada uno de los cuerpos celestes de la muestra (ver Tabla 5.1), implementando el método de tránsito, aplicando además ventanas de observación de diferente tamaño y resolución alrededor de estos. Por otro lado, dado el poco tiempo de integración que implica realizar la observación a través de escaneos, solo se implementó este método en aquellos objetos de mayor emisión. Los resultados obtenidos se muestran a continuación.

Sol

Al apuntar el Sol implementando la observación matricial se logra una detección efectiva. Como se muestra en las Figuras 7.1 y 7.2, es posible realizar la observación de este objeto implementando incluso la ventana de máximo tamaño, gracias a su ancho angular y magnitud de radiación. Por otro lado, al ejecutar la técnica de escaneo se observa nuevamente un aumento considerable en la respuesta del radiorreceptor cuando este pasa por su lóbulo principal. Al observar la Figura 7.3, se aprecia la respuesta generada al realizar el escaneo durante media hora, permitiendo visualizar el movimiento descrito por el Sol cuando este recorre la eclíptica. Finalmente, la detección de medio tránsito y un tránsito completo se aprecia en la Figura 7.4, donde se observa un cambio claro de casi $9\text{ dB}\mu\text{V}$ en la salida del radiorreceptor.

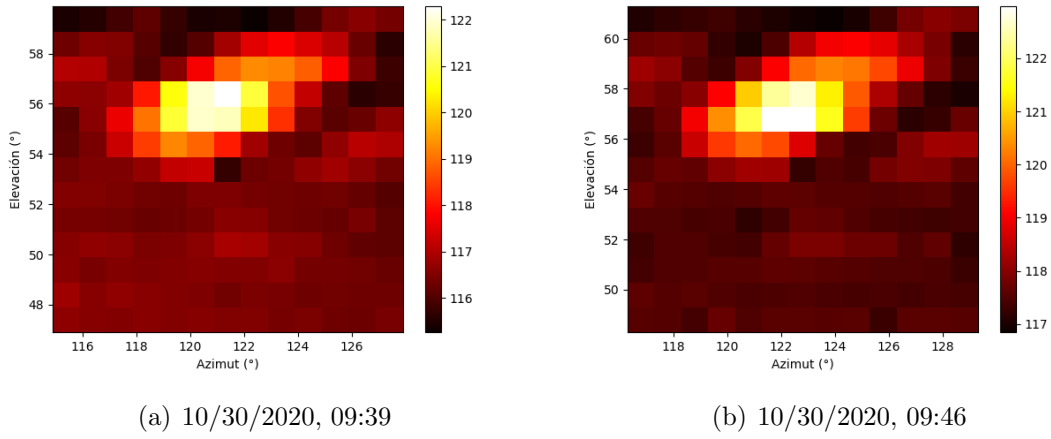


Figura 7.1: Ventana de observación de $12^\circ \times 12^\circ$ y pasos de 1° alrededor del Sol.

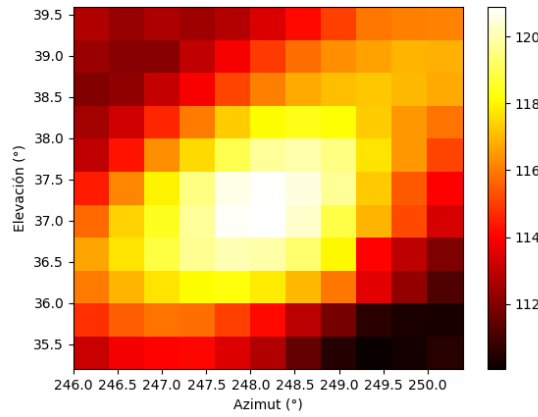


Figura 7.2: Ventana de observación de $4^\circ \times 4^\circ$ y pasos de 0.4° alrededor del Sol (10/29/2020, 15:03).

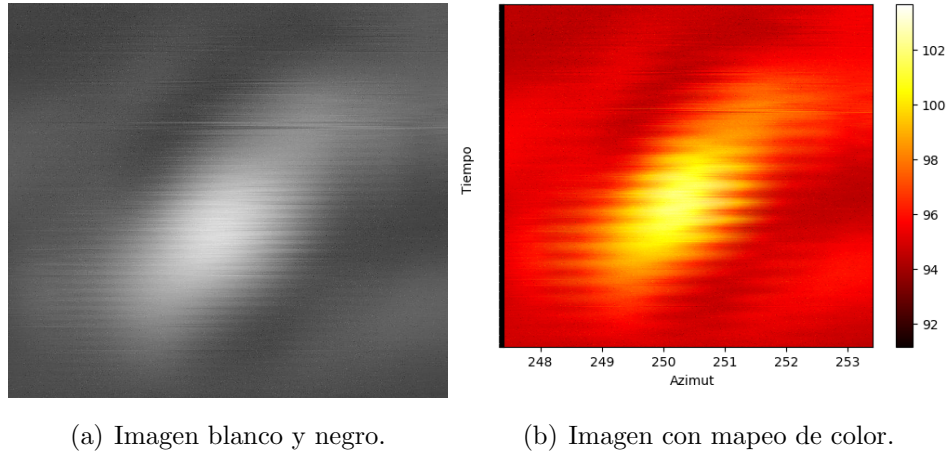


Figura 7.3: Escaneo sobre el sol el 11/05/2020 desde las 15:25, durante 30 minutos.

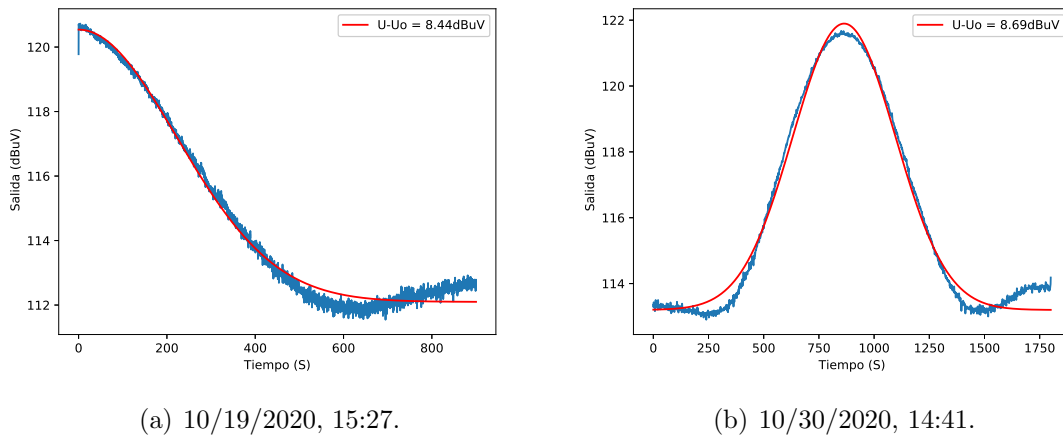


Figura 7.4: Detección de tránsito solar durante 15 y 30 minutos respectivamente.

Luna

Al apuntar el radiotelescopio hacia la Luna, se logra la detección al implementar matrices de observación de un tamaño de $4^\circ \times 4^\circ$ como se observa en las Figuras 7.5 y 7.6. Por otro lado, la detección de un escaneo se aprecia en la Figura 7.7. Finalmente, al detectar medio tránsito de la Luna se aprecia un cambio en la respuesta del radiorreceptor de hasta $0.84 \text{ dB}\mu\text{V}$ (ver Figura 7.8).

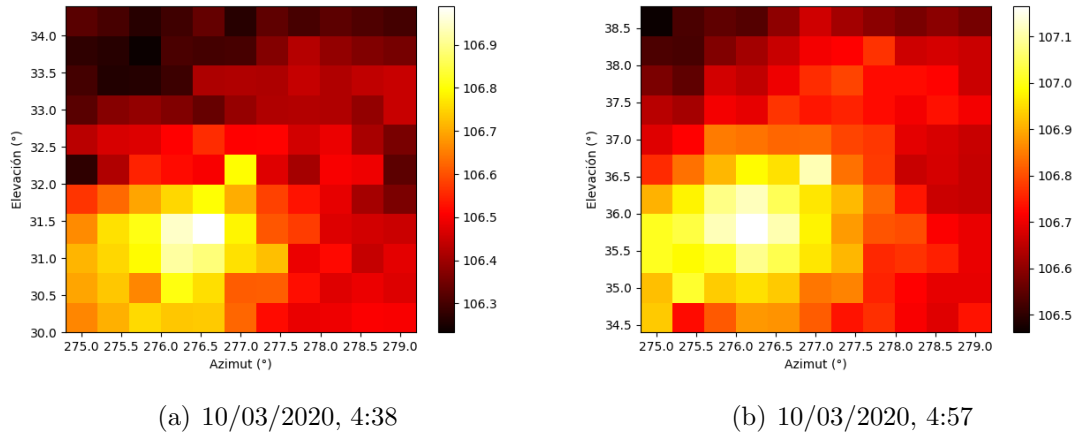


Figura 7.5: Ventana de observación de $4^\circ \times 4^\circ$ y pasos de 0.4° alrededor de la Luna.

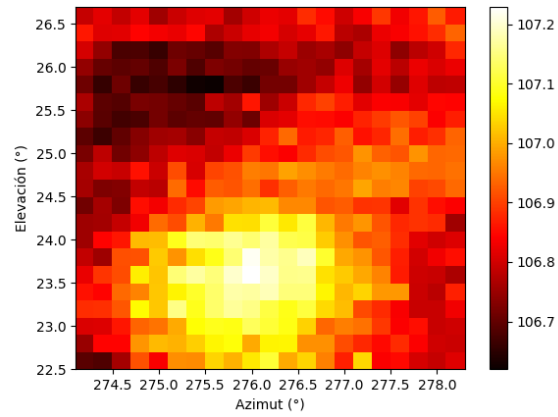
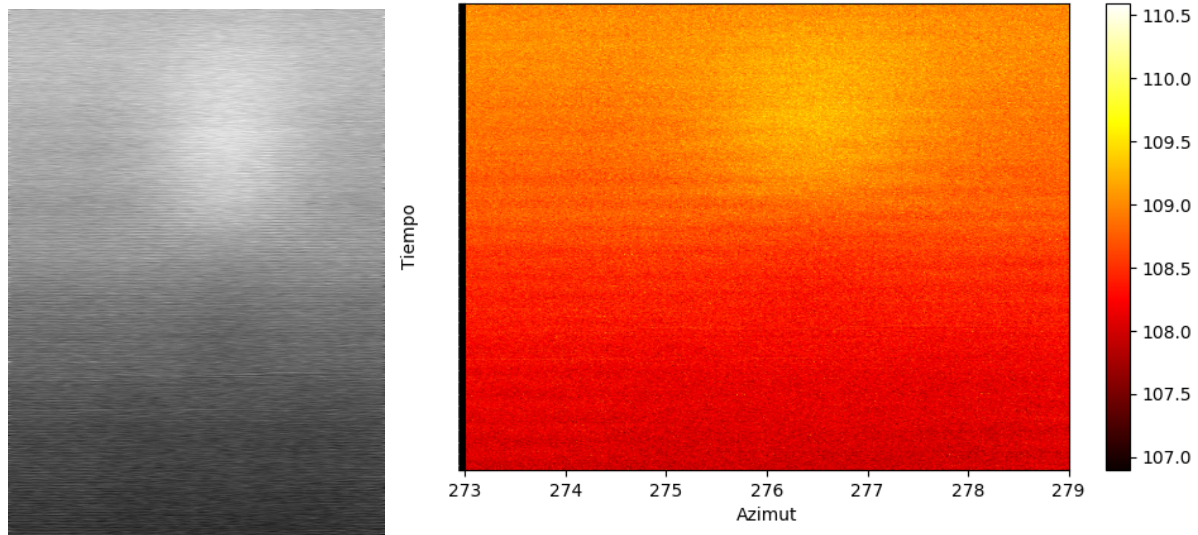
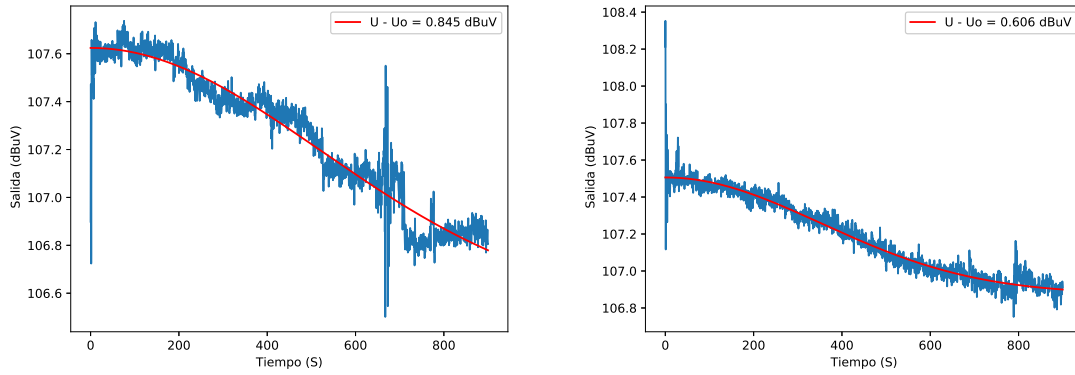


Figura 7.6: Ventana de observación de $4^\circ \times 4^\circ$ y pasos de 0.2° alrededor de la Luna (10/03/2020, 05:27).



(a) Imagen blanco y negro.

(b) Imagen con mapeo de color.

Figura 7.7: Escaneo sobre la luna el 10/03/2020 desde las 05:15, durante 30 minutos.

(a) 10/03/2020, 4:56.

(b) 10/03/2020, 5:16.

Figura 7.8: Detección de medio tránsito lunar durante 15 minutos.

Centro galáctico

Para la detección del centro galáctico se ejecutó la ventana de observación matricial más ancha alrededor de Sgr A. Como se observa en la Figura 7.9, no es posible identificar un único objeto emisor, por el contrario, se observan cambios en la intensidad por zonas, lo que significa que el centro galáctico no puede ser observado en su totalidad con esta técnica. Teniendo en cuenta esto, se ejecutaron tránsitos completos al apuntar Sgr A de hasta 3 horas, como se observa en la Figura 7.10. En la imagen se observa un tránsito efectivo en la gráfica izquierda, mientras que, en la gráfica derecha se observa el paso de una nube de lluvia, generando una caída en el voltaje del sistema cuando cruza completamente. Aunque las condiciones climáticas fueron diferentes, se evidencia la poca afectación de las variables meteorológicas en la efectividad del sistema, ya que, el cambio en respuesta al detectar la fuente de radiación sigue siendo evidente.

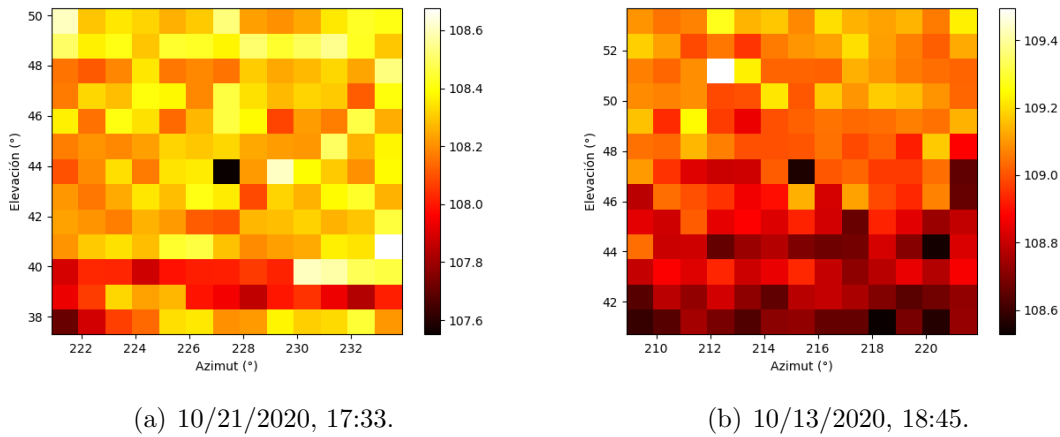
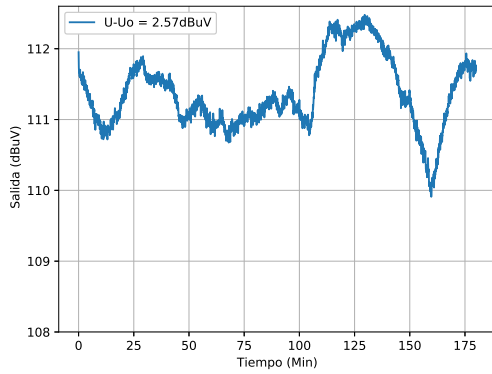
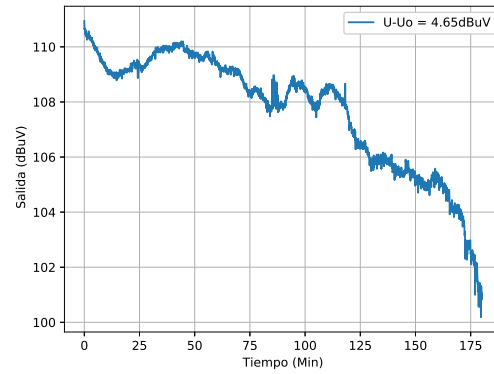


Figura 7.9: Ventana de observación de $12^\circ \times 12^\circ$ y pasos de 1° alrededor del centro galáctico.



(a) 10/23/2020, 19:45.

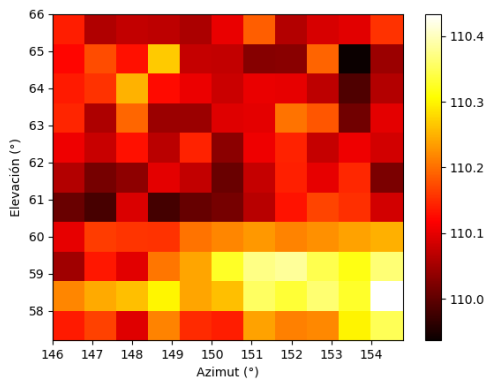


(b) 10/26/2020, 19:55.

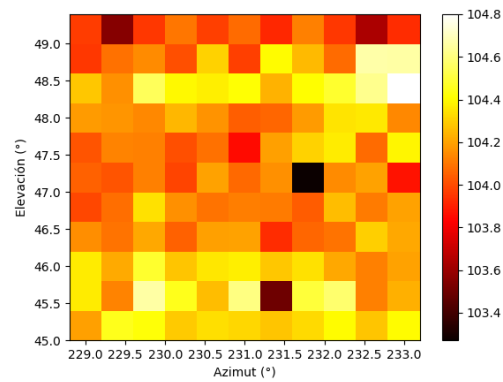
Figura 7.10: Detección de tránsito galáctico durante 3 horas.

Saturno

Una vez descartados los objetos de mayor tamaño angular, se presentan los resultados al observar aquellos objetos de la muestra con una menor intensidad de flujo. Para la detección de Saturno, se implementaron ventanas de observación de $4^\circ \times 4^\circ$ y pasos de 0.4° con el objetivo de detectar el objeto de interés en la zona observada. Por otro lado, se ejecutaron tránsitos de 15 minutos sobre el planeta.

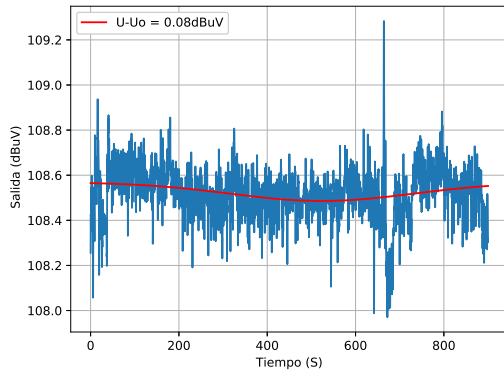


(a) 10/13/2020, 17:11.

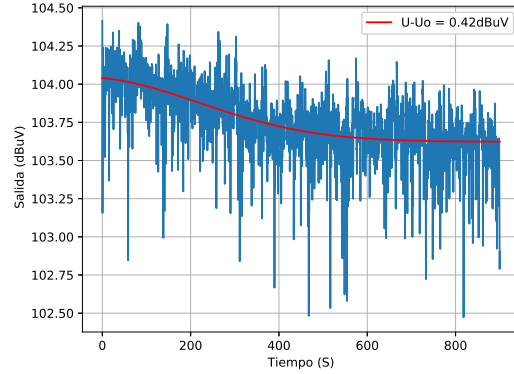


(b) 10/19/2020, 20:16.

Figura 7.11: Ventana de observación de $4^\circ \times 4^\circ$ y pasos de 0.4° alrededor de Saturno.



(a) 10/15/2020, 20:58.

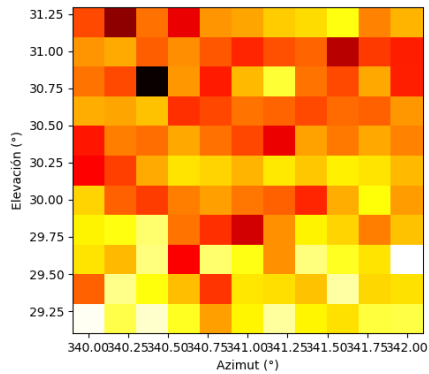


(b) 10/19/2020, 20:40.

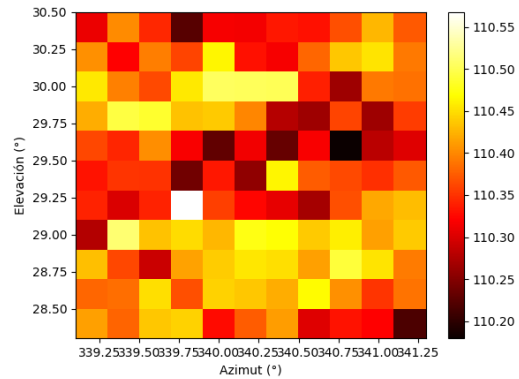
Figura 7.12: Tránsitos de 15 min sobre Saturno.

Para la observación de los objetos de espacio profundo, se realizaron ventanas de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° , además de tránsitos con una duración de 20 minutos. Los resultados obtenidos se muestran a continuación.

Cassiopeia A

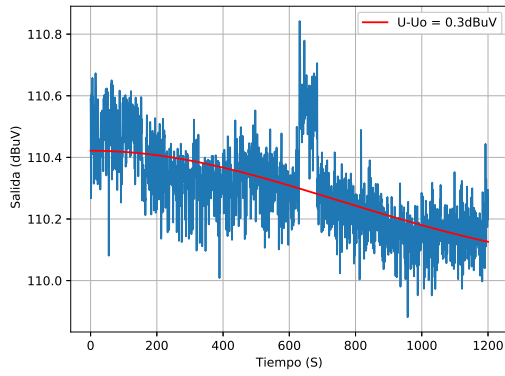


(a) 10/16/2020, 23:54.

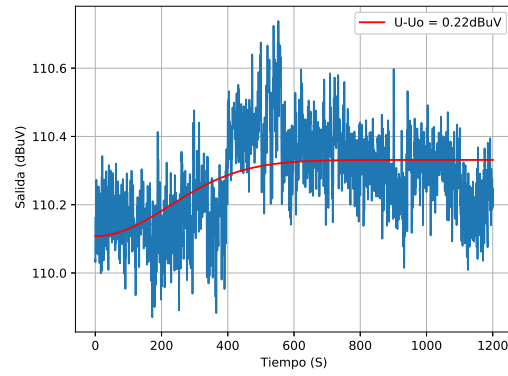


(b) 10/17/2020, 00:00.

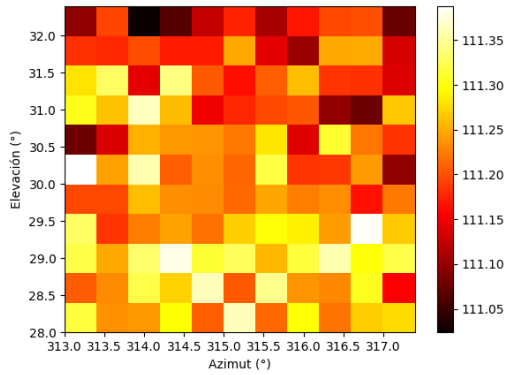
Figura 7.13: Ventana de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° alrededor de Cassiopeia A.



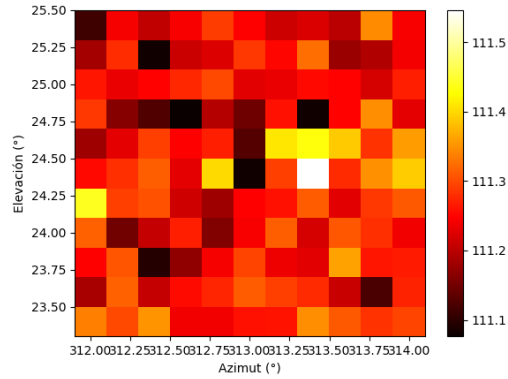
(a) 10/17/2020, 00:53.



(b) 10/17/2020, 1:38.

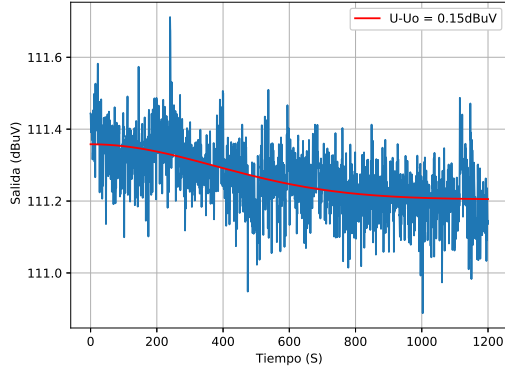
Figura 7.14: Tránsitos de 20 min sobre Cassiopeia A.**Cygnus A**

(a) 10/16/2020, 21:52.

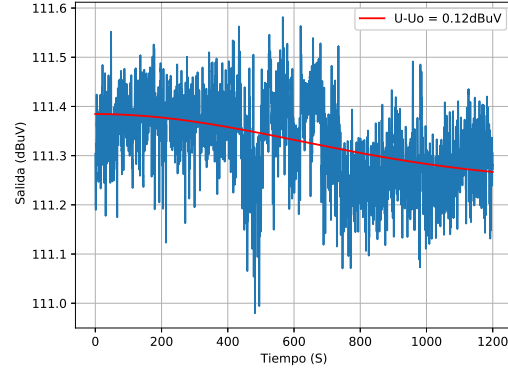


(b) 10/16/2020, 22:24.

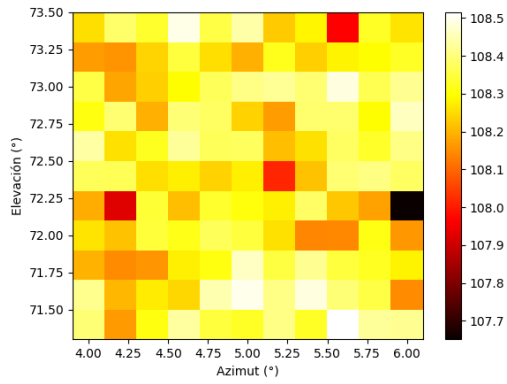
Figura 7.15: Ventana de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° alrededor de Cygnus A.



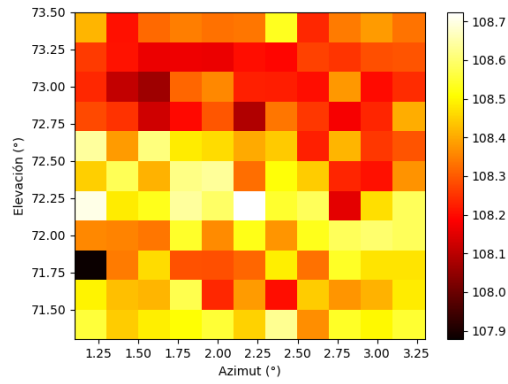
(a) 10/16/2020, 22:22.



(b) 10/16/2020, 22:47.

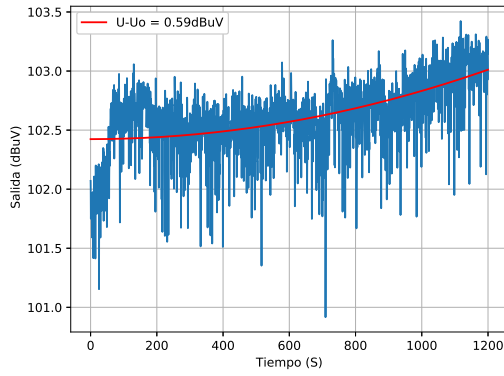
Figura 7.16: Tránsitos de 20 min sobre Cygnus A.**Taurus A**

(a) 10/17/2020, 3:47.

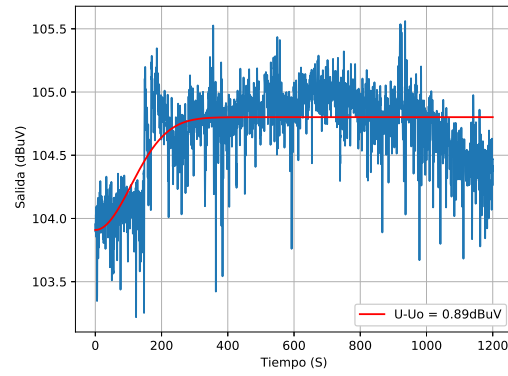


(b) 10/17/2020, 3:50.

Figura 7.17: Ventana de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° alrededor de Taurus A.



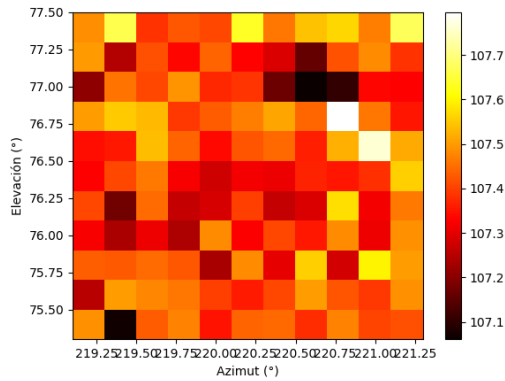
(a) 10/17/2020, 6:40.



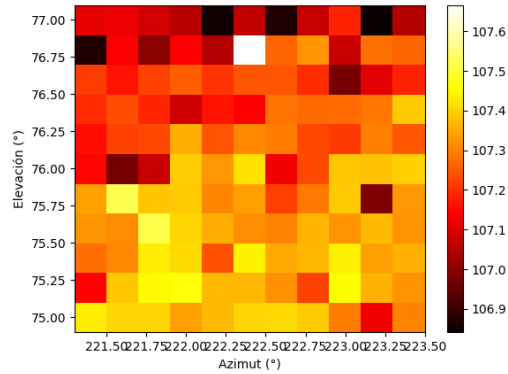
(b) 10/17/2020, 8:06.

Figura 7.18: Tránsitos de 20 min sobre Taurus A.

Nebulosa de Orión



(a) 10/17/2020, 4:28



(b) 10/17/2020, 04:31

Figura 7.19: Ventana de observación de $2^\circ \times 2^\circ$ y pasos de 0.2° alrededor de la Nebulosa de Orión.

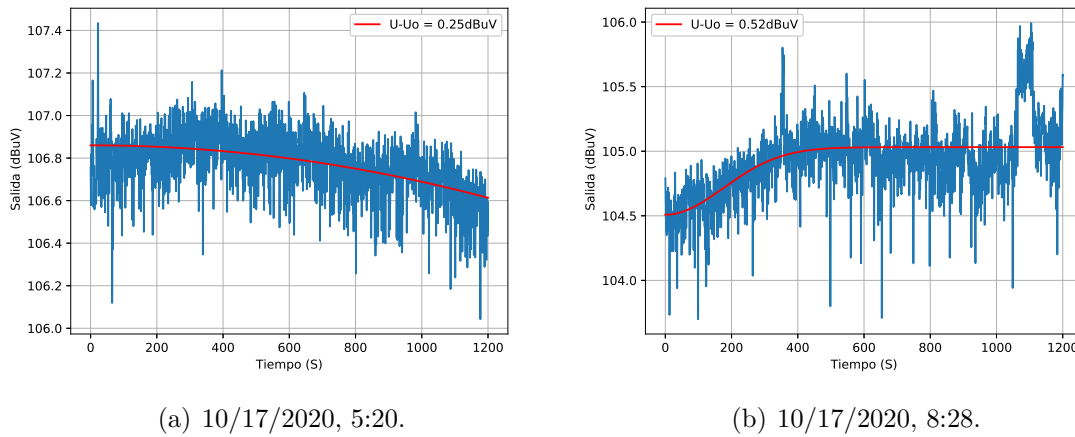


Figura 7.20: Tránsitos de 20 min sobre la Nebulosa de Orión.

Como se muestra en las Figuras 7.11, 7.13, 7.15, 7.17, 7.19, al implementar las observaciones con la ventana matricial no es posible detectar ninguno de los objetos, visualizando únicamente ruido de fondo sin evidenciar una detección positiva. Por otro lado, al ejecutar el método de tránsito, se obtienen los resultados de las Figuras 7.12, 7.14, 7.16, 7.18, 7.20, donde solo es posible visualizar cambios generados por el paso de nubes de lluvia. Por consiguiente, solo se logran detecciones positivas al apuntar el Sol, la Luna y el centro galáctico.

7.1.2 Flujo y temperatura de brillo

Para el caso del Sol y la Luna se obtuvo un cambio en la respuesta considerable, por lo que se evidencia una detección positiva. Ahora, utilizando la expresión de la ecuación (5.3.7), se calcula el flujo de ambos objetos, obteniendo los resultados de la Tabla 7.1. Cabe resaltar que, para observaciones realizadas durante el día se estimó una temperatura del sistema de 204.3 K, mientras que, para observaciones nocturnas, se estima una temperatura del sistema de 108.6 K.

Objeto	ΔU_{dB} (dBuV)	Flujo (Jy)
Sol	8.44	2.594.800,8
Luna	0.84	49.200,2

Tabla 7.1: Flujo de radiación de las radiofuentes observadas.

En el caso particular de la Vía Láctea, no es posible identificar una única fuente de temperatura. En consecuencia, procedimos a analizar los rangos de flujo observado durante el tránsito captado el 23 de octubre desde las 4:45 pm, identificando los puntos mínimo y máximo de detección con una respuesta al cielo de fondo de 110.2 dBuV.

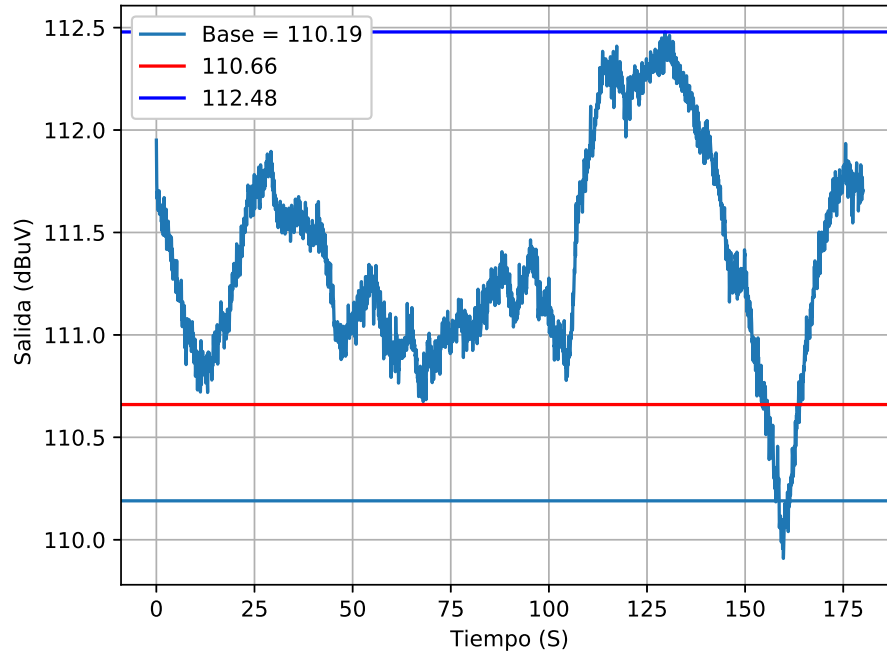


Figura 7.21: Punto de máxima y mínima de detección respecto al cielo de fondo

Tomando los datos de la Figura 7.21, se puede calcular el aumento de la respuesta respecto al cielo de fondo para el punto mínimo (ΔUdB_1) y máximo (ΔUdB_2).

$$\begin{aligned}
 \Delta UdB_{min} &= 0.47dBuV \\
 S_{min} &= 49501.84Jy \\
 \Delta UdB_{max} &= 2.29dBuV \\
 S_{max} &= 300723.04Jy
 \end{aligned} \tag{7.1.1}$$

Con los valores obtenidos, se observa como el flujo de radiación detectado por la Vía Láctea varía desde los 49500 Jy hasta los 300723 Jy durante la detección del tránsito. Así, la magnitud de variación del flujo ΔS , deja en evidencia las múltiples fuentes de radiación térmica circundantes con el centro galáctico.

$$\Delta S = 251222.04Jy$$

De las radiofuentes observables se llevará a cabo el cálculo de la temperatura de brillo de aquellas con un ancho angular definido, aplicando la ecuación (5.3.5). Para el Sol, se realizó el procedimiento durante varios días, como se muestra en la Tabla 7.2. Durante la observación solar, se estimó una temperatura de brillo promedio observada por la antena de 1068.51 K,

mientras que, para la Luna, se observó una lectura de temperatura de 242.5 K. Así, el ancho angular y la temperatura de brillo de los objetos observados por la antena se presenta en la Tabla 7.3. Estos valores se compararon con las observaciones a 11 GHz realizadas en el Observatorio Astronómico de Strasbourg¹, evidenciando que se encuentra en el rango de temperatura permitido. Dados los cambios en la intensidad de radiación emitida por el Sol durante el ciclo solar, la temperatura de brillo puede variar desde los 5000 K hasta 15000 K.

Fecha	Temperatura de brillo (K)
10/29/2020	11014.92
10/29/2020	10944.59
10/30/2020	8340.36
10/30/2020	9093.75
10/30/2020	10285.97
11/03/2020	8813.83
11/03/2020	11258.72
11/03/2020	10795.96

Tabla 7.2: Temperatura de brillo de Sol.

Objeto	Ancho angular (°)	Temperatura de brillo (K)	Rango (K)
Sol	0.524	10068.51	6000 - 15000
Luna	0.488	242.5	180 - 250

Tabla 7.3: Temperatura de brillo de radiofuentes observadas.

¹<https://portia.astrophysik.uni-kiel.de/koeppen/10GHz/>

7.2 Capacidades didácticas del radiotelescopio

La estrategia didáctica se aplicó para 15 estudiantes divididos en 4 grupos, 3 de 4 y uno de 3, pertenecientes a programas de pregrado en ingeniería de la Universidad Tecnológica de Pereira. Con el fin de contrastar los resultados de la ZDP y la ZDPr se dividen las preguntas realizadas en el grupo de discusión en tres categorías: Astronomía, Radiación solar y Radioastronomía y Anetas. Antes de iniciar la práctica de laboratorio se aplicó el grupo de discusión con los participantes, recopilando las respuestas de los estudiantes. Posteriormente, se aplica la estrategia didáctica, suministrando la guía de laboratorio y las herramientas necesarias para la ejecución de la práctica. Al finalizar se aplica nuevamente el grupo de discusión, recopilando las respuestas acertadas por los estudiantes. A continuación se muestra el porcentaje de estudiantes cuya respuesta fue acertada antes y después de aplicar el instrumento para la enseñanza.

Pregunta	Respuesta esperada	Respuestas acertadas	
		ZDR	ZDPr
¿Bajo sus palabras, para usted qué es la astronomía?	Ciencia enfocada en el estudio de los astros	66.7%	86.7%
¿De qué formas cree usted que se pueden estudiar los cuerpos celestes y fenómenos astronómicos	A través de los diferentes tipos de radiación provenientes del objeto	40%	66.7%
¿En los estudios enfocados hacia la astronomía qué instrumentos se observan frecuentemente?	Radiotelescopios	46.7%	93.3%
Normalmente la astronomía óptica se ve limitada por las condiciones climáticas ¿Cree usted que existe una vía alternativa para el estudio de los cuerpos celestes que evite esta dificultad?	A través de la radioastronomía	46.7%	86.7%

Tabla 7.4: Resultados Astronomía.

Pregunta	Respuesta esperada	Respuestas acertadas	
		ZDR	ZDPr
El Sol es una estrella que emite radiación constantemente ¿Existe algún tipo de información que podamos extraer de la radiación proveniente de esta estrella?	Si, información como: Temperatura de sus capas, comportamiento, actividad solar y composición	66.7%	100%
¿Considera usted que la actividad del Sol afecta en alguna medida las actividades de los humanos en la Tierra	Si, grades cantidades de energía proveniente del Sol que llegan a la Tierra pueden afectar las comunicaciones	40%	100%
¿De qué formas se puede estudiar el Sol considerando que es un objeto que se encuentra muy alejado de nosotros?	A través de la radiación proveniente de él	60%	86.7%

Tabla 7.5: Resultados Radiación solar.

En las categorías de Astronomía y Radiación solar, en la ZDP, se evidencia que un porcentaje alto de los participantes tiene conocimiento de enfoques de la y los fenómenos causados por la radiación solar. Por otro lado, las ventajas presentes en la radioastronomía y los instrumentos utilizados en esta se tiene poco claro. En la ZPr se evidencia se observa un amento significativo en la comprensión de estos conceptos, alcanzando a acabarcar casi el 100% de los participantes.

Pregunta	Respuesta esperada	Respuestas acertadas	
		ZDR	ZDPr
Las antenas son instrumentos utilizados actualmente en los sistemas de comunicaciones ¿Qué características conoce de estos dispositivos?.	Captan y emiten ondas, su forma y diseño depende de la longitud de onda a observar.	73.3%	93.3%
¿Cuáles cree que podrían ser los limitantes de la tecnología basada en el uso de antenas?.	Grandes tamaños por la baja resolución.	40%	53.3%
Según sus conocimientos ¿Qué entiende por radioastronomía?.	Estudio de los cuerpos celestes a través de las ondas electromagnéticas en la región de radio	53.3%	93.3%

Tabla 7.6: Resultados Radioastronomía y Antenas.

En la Categoría Radioastronomía y Antenas, se encontró, en la ZDR, que cerca de un 50% de los estudiantes tenía conocimiento previo del tema, mientras que ninguno evidenciaba un contacto anterior con el tipo de instrumentación utilizada. Al involucrar la estrategia didáctica para la enseñanza de la radiastronomía, cerca del 80% de los participantes adquiere una mayor comprensión de los conceptos de emisión de radioondas y el funcionamiento de los radiotelescopios en la ZDPr. Además, se alcanza una cobertura del 100% de participantes que logran interactuar de manera práctica con los instrumentos como se aprecia en la Figura 7.22.



Figura 7.22: Desarrollo de la práctica.

Capítulo 8

Conclusiones

Al realizar la adquisición de la temperatura del sistema y el ancho de haz de la antena a distintas horas del día, es posible visualizar la variación de estos parámetros bajo distintas condiciones. Como primera observación, se evidencia que la temperatura del sistema presenta sus valores más bajos durante las horas de la mañana y la noche, presentando un aumento constante a medida que desciende la tarde, disminuyendo en la noche. Por otro lado, la mitad del ancho de haz varía entre 1° y 1.3° , dependiendo de la elevación del objeto, consecuencia de la refracción atmosférica. Para evitar cambios abruptos en este valor se recomienda realizar la observación de objetos con una altitud superior a los 20° .

Tras realizar el proceso de calibración del plato reflector y apuntado se evidencia la factibilidad de observación del Sol y la Luna, por lo que es posible medir su temperatura de brillo y flujo de radiación a 11.2 GHz. Por otro lado, se logra la detección del centro galáctico y sus proximidades, siendo necesaria la ejecución de tránsitos de tres horas. Dada la duración del proceso y la identificación de distintos picos de emisión, la temperatura de brillo no pudo ser calculada, pero se pudo determinar el cambio nominal de flujo de radiación durante la adquisición. Finalmente, dado el imperceptible cambio en respuesta que generan las demás radiofuentes, estas no pueden ser observadas. Por consiguiente, el potencial científico del instrumento se centra en el cálculo del flujo y temperatura de brillo de la Luna y el Sol, además de los cambios de flujo al detectar el centro galáctico, donde abundan las radiofuentes térmicas.

El Sol destaca por ser el objeto que genera la mejor respuesta en el radiorreceptor. Dado que su temperatura de brillo se encuentra ligada a la intensidad de radiación emitida, uno de los usos potenciales del radio, para fines científicos, es el seguimiento del ciclo solar (Ho et al., 2008). Al calcular la temperatura de brillo diariamente, durante largos periodos de tiempo, es posible alimentar una robusta base de datos que permita visualizar los cambios

en la intensidad de radiación percibida durante la evolución del ciclo solar.

Con el fin de mejorar las capacidades del radiotelescopio, se recomienda ampliar el diámetro del plato reflector de 1.2 m a 5 m. Además, para aprovechar al máximo el diámetro de la antena, es necesario que su área iluminada sea lo más cercana a su área física, por lo que su eficiencia debe ser mayor a 0.6. Para lograr una buena iluminación es necesario utilizar un material sólido para el plato reflector, además de mantener la superficie lo más limpia posible. La modificación de estos parámetros permitirá al radirreceptor aumentar considerablemente su ganancia, ya que, el nuevo diámetro de 5 m proporciona un ancho de haz de 0.36° , habilitando la detección de fuentes con un flujo de radiación mínima 200 Jy .

Tras el estudio realizado en este proyecto se logró un recopilado de diferentes consideraciones que se deben tener presentes durante el manejo del radiotelescopio, esto para su correcta operación y una mejor adquisición de datos.

- Tener presente la ubicación de los satélites geoestacionarios que emiten en la banda KU para evitar interferencia con el estudio de los objetos.
- Convertir de sexagesimal a decimal las posiciones de los objetos presentadas en Stellarium. Esto para ingresarlas al Software CRT-UTP.
- Evitar las ventanas de observación para la calibración a elevaciones menores a 15° y mayores a 75° .
- Implementar ventanas de observación con una relación tamaño-resolución baja para objetos que se desplazan sobre el cielo.
- Tener presente el error al cambiar de hemisferio, como también las consideraciones que se plantean en el apartado de calibración.

La estrategia didáctica generó un escenario de aprendizaje vivencial, donde el estudiante se relaciona con los instrumentos astronómicos. Paralelo a la práctica se desarrollan las habilidades de correlación e interpretación, logrando a través de una experiencia didáctica la aclaración de conceptos relacionados con la radioastronomía y el funcionamiento de las antenas. Una alternativa para la ejecución de la práctica se encuentra en calcular la temperatura de brillo de la Luna. Aquí, los estudiantes pueden observar el comportamiento de la respuesta del receptor al apuntar la Luna, un objeto cuya intensidad radiación es consecuencia de la reflexión de ondas electromagnéticas en su superficie. Esta nueva práctica de laboratorio se recomienda como trabajo a futuro.

Bibliografía

- Bhatia, R. S., Marti-Canales, J., De Matos, C., Fritzsche, B., Haiduk, F., Knoechel, U., y Furse, C. (2006). A simple radio telescope operating at ku band for educational purposes. *IEEE Antennas and Propagation Magazine*, 48(5).
- Castillo, Y. (2010). Diseño de un raditelescopio para actividades de investigación del departamento de astronomía y astrofísica de la FACES/UNAH. *Revista Ciencias Espaciales*, 2(1):40–53.
- Chanka, N. (2018). Development of an off the shelf methanol maser telescope. Master's thesis, School of Physics , Astronomy and Mathematics.
- de Parter, I. y S. Kurth, W. (2014). Chapter 52 - the solar system at radio wavelengths. In Spohn, T., Breuer, D., y Johnson, T. V., editors, *Encyclopedia of the Solar System (Third Edition)*, pages 1107 – 1132. Elsevier, Boston, third edition edition.
- de Pater, I., Butler, B., Sault, R. J., Moullet, A., Moeckel, C., Tollefson, J., de Kleer, K., Gurwell, M. A., y Milam, S. (2019). Potential for Solar System Science with the ngVLA: the Giant Planets. *Bulletin of the American Astronomical Society*, 51(3):19.
- E. Yousif, M. (2014). The electromagnetic radiation mechanism. *International Journal of Fundamental Physical Sciences (IJFPS) ISSN:2231-8186*, Vol 4:72–79.
- Falcienelli, F. (2014). Spider230 experiences: Cassiopeia a transit at 11 ghz. <http://radioastrolab.com/experiments-at-11-2-ghz-with-spider230-radio-telescope-cassiopeia-a>. Accessed: 2019-10-15.
- Figueredo, P. S. y Hernández, S. V. (2009). Prototipo de receptor radioastronómico de nueva generación en la banda ku del espectro electromagnético. *Ciencias de la Tierra y el Espacio*, 10:87–94.
- Galvis Rodriguez, H., Quintero Salaza, E., y Cardona, L. (2016). Development of a magnetic loop antenna for the detection of jovian radiowaves at 20.1 mhz. *TECCIENCIA*, 11(20):41–46.

- García Ramírez, D. E. (2019). Evaluación de la habilidad científica de análisis alcanzada con la implementación de una estrategia de enseñanza socioconstructivista de cráteres lunares. Master's thesis, Maestría en enseñanza de la Física.
- Gaylard, M. (2012). *Radio Astronomy with a Satellite Dish*. South African Radio Astronomy Observatory.
- Gómez Tornero, J. L., Cañete Rebenaque, D., Quesada Pereira, F. D., y Álvarez Melcón, A. (2010). Development of a small radio telescope at the technical university of cartagena. In *EDUCON*.
- Gulkis, S. y de Pater, I. (2003). Radio astronomy, planetary. In Meyers, R. A., editor, *Encyclopedia of Physical Science and Technology (Third Edition)*, pages 687 – 712. Academic Press, New York, third edition edition.
- Hiep, N., Anh, P., Diep, P., Pham Ngoc, D., Hoai, D., Nhung, P., Thao, N., y Darriulat, P. (2012). The vatly radio telescope. *Communications in Physics*, 22:365–374.
- Higuita, J., Salazar, E., y Ramírez, I. (2019). Development of a radio telescope for the observation of celestial bodies in the ku band. *Journal of Physics: Conference Series*, 1247:012034.
- Ho, C., Slobin, S., Kantak, A., y Asmar, S. (2008). Solar brightness temperature and corresponding antenna noise temperature at microwave frequencies. *Interplanetary Network Progress Report*.
- Holleman, I. y Beekhuis, H. (2004). Weather radar monitoring using the sun. Technical report, Royal Dutch Meteorological Institute.
- Holleman, I., Huuskonen, A., Kurri, M., y Beekhuis, H. (2010). Operational monitoring of weather radar receiving chain using the sun. *Journal of Atmospheric and Oceanic Technology - J ATMOS OCEAN TECHNOL*, 27.
- Jaramillo Alvarado, A., Galvis Rodriguez, H., y Quintero Salazar, E. (2017). Space weather monitoring from colombia through higher order butterworth radio-receivers. *Ingeniería*, 22(1).
- Mandal, P., Agarwal, D., Kumar, P., Yelikar, A., Soni, K., y Krishna, V. (2016). Everyday radio telescope.
- Manisha, A. (2016). Design of a radiotelescope at a frequency band of 10 to 12 ghz. *International Journal of Advanced and Latest Research in Engineering Science and Technology*, 1.

- Mendoza Torres, E. (2003). El radiotelescopio y la observación celeste. *Elementos*, 2(12):92–98.
- Montoya, L. y del Mar, M. (2011). Enseñanza y divulgación de las ciencias : Introducción a la radioastronomía. *Revista 100cias@uned*, 4:143–152.
- Muth, X., Schneebeli, M., y Berne, A. (2011). A sun-tracking method to improve the pointing accuracy of weather radar. *Atmospheric Measurement Techniques Discussions*, 4:547–555.
- O’Neil, K. (2001). Single dish calibration techniques at radio wavelengths. In *ASP Conference Series*.
- Pater, I., Butler, B., Sault, R., Moullet, A., Moeckel, C., Tollefson, J., Kleer, K., Gurwell, M., y Milam, S. (2018). Potential for solar system science with the ngvla.
- R. Hoobi, M. y M. Abood, K. (2019). Calibration of a three meter small radio telescope in baghdad university using the sun as reference source. *Iraqi Journal of Science*, 60(1):171–177.
- Raulin, J.-P. y Pacini, A. (2005). Solar radio emissions. *Advances in Space Research*, 35:739–754.
- Saje, T. y Vidmar, M. (2017). A compact radio telescope for the 21 cm neutral-hydrogen line. *Journal of Microelectronics, Electronif Components and Materials*, 47(2):113–128.
- Tinti, M. (2013). Construction of a 12 ghz total power radio telescope for teaching pruposes, suitable for noisy environments, using satellite tv devices. *Progress In Electromagnetics Research*, 37:159–170.
- Uwe, K. y Frank, H. (2005). Calibration report. Technical report, Fraunhofer Institute IIS.

Anexo A

Estrategia didáctica

A.1 Observación participante

Fecha:_____ Lugar:_____ Número de estudiantes:_____

Nombre de la actividad:

Objetivos:

Pregunta esencial:

¿_____?

Descripción de la actividad:

Elemento de análisis:

Observación del elemento de análisis:

Conclusiones:

Recomendaciones:

A.2 Grupo de discusión

Fecha:_____ Lugar:_____ Hora:_____

Conducta de entrada: Se iniciará con una actividad introductoria enfocada en disminuir la tensión en los estudiantes, amenizando la futura participación de cada uno.

Introducción: La intervención tiene como objetivo definir el contexto de desarrollo en el grupo de estudiantes. En una etapa inicial, se investigará la relación de los integrantes de grupo respecto a la zona de desarrollo real (ZDR), definiendo los conocimientos previos. En una segunda etapa la investigación se enfocará en la zona de desarrollo potencial (ZDP), donde se analizará el nivel de argumentación y apropiación del conocimiento de los estudiantes en el momento posterior a la intervención.

Objetivo: Identificar los conceptos físicos implicados en la radioastronomía presentes en una muestra de estudiantes del pregrado en Ingeniería física de la Universidad Tecnológica de Pereira.

Dinámica del grupo de discusión: Se inicia la actividad dando las pautas a seguir durante el desarrollo de la misma. El respeto a la palabra y opinión de los participantes se enfatizará antes de iniciar el grupo de discusión. Se procede entonces con el planteamiento de las preguntas y se inician las intervenciones.

- Bajo sus palabras, para usted ¿Qué es la astronomía?.
- ¿De qué formas cree usted que se pueden estudiar los cuerpos celestes y fenómenos astronómicos?.
- ¿En los estudios enfocados hacia la astronomía qué tipo de instrumentos se observan frecuentemente?.
- Normalmente la astronomía óptica se ve limitada por las condiciones climáticas. ¿Cree usted que existe una vía alterna para el estudio de los cuerpos celestes que evite esta dificultad?.
- El sol es una estrella que emite radiación constantemente ¿Existe algún tipo de información que podamos extraer de la radiación proveniente de esta estrella?.
- ¿Considera usted que la actividad del Sol afecta en alguna medida las actividades de los humanos en la Tierra?.

- ¿De qué formas se puede estudiar el Sol considerando que es un objeto que se encuentra muy alejado de nosotros?
- Las antenas son instrumentos utilizados actualmente en los sistemas de comunicaciones ¿Qué características conoce de estos dispositivos?.
- ¿Cuáles cree que podrían ser los limitantes de la tecnología basada en el uso de antenas?.
- Según sus conocimientos ¿Qué entiende por radioastronomía?.
- Tiene algún conocimiento o idea sobre los siguientes fenómenos: Longitud de onda, Emisión de ondas, Radiación de Cuerpo Negro o Temperatura de Brillo. Si le es similar alguno de los anteriores descríbalos con sus palabras de forma concisa.

A.3 Guía aplicativa

LABORATORIO DE RADIOASTRONOMÍA

PRÁCTICA: TEMPERATURA DE BRILLO DEL SOL EN 11 GHz.

1. Resumen

En la práctica de laboratorio se implementa el radiotelescopio instalado en el Observatorio Astronómico de la Universidad Tecnológica de Pereira, el cual se encuentra motorizado en azimut y altitud, facilitando el proceso de apuntado. Durante la práctica se definen distintos parámetros de funcionamiento del instrumento y una metodología para determinar la temperatura media de brillo del sol a partir de observaciones realizadas en 11 GHz. En esta frecuencia las ondas emitidas por la radiación solar se caracterizan por ser provenientes de una zona ubicada entre la corona solar baja y la zona de transición, donde las temperaturas del plasma alcanzan valores entre los 5000 y 20000 K aproximadamente.

2. Objetivos

- Conocer los aspectos generales de funcionamiento del radiotelescopio OAUTP.
- Familiarizarse con los conceptos físicos asociados a la radioastronomía y su uso para la observación de la actividad solar en la banda KU.
- Determinar la temperatura de brillo del sol en las proximidades de la corona solar baja.

3. Fundamentos teóricos

3.1. Radioastronomía y radiación solar

El espectro electromagnético está compuesto por un amplio conjunto de ondas, aún así sólo una fracción de ellas es observable para el ojo humano, por lo que el estudio de los cuerpos celestes se ve enfocado en el análisis de la región de espectro óptico. Hoy en día es bien sabido que los cuerpos celestes emiten en un rango más amplio que el visible, incluyendo las ondas de radio, las cuales son captables a través de dispositivos llamados radiotelescopios. El estudio de los fenómenos apreciables a través de esta nueva ventana de observación se define como radioastronomía, una ciencia fundamentada en la física de ondas electromagnéticas provenientes de los astros en la región de radio del espectro. La banda Ku se encuentra ubicada, aproximadamente, en la unión de las ondas infrarrojas y las ondas de radio (12 - 18 GHz) lo cual indica que la radiación electromagnética ingresa hasta la superficie terrestre. Las ondas incluidas en este rango de frecuencia pueden interactuar en menor medida con las partículas de agua presentes en el ambiente, por lo que las condiciones climáticas dejan de

ser un problema para captarlas.

El sol es un objeto celeste cuyos fenómenos pueden ser observados implementado diferentes técnicas, siendo la radioastronomía una de las más utilizadas, por lo que abre una amplia ventana para los estudios en astrofísica a diferentes longitudes de onda. A 11 GHz, la radiación proveniente del sol es de clasificación térmica, ya que, el mecanismo de emisión se caracteriza por ser radiación de cuerpo negro, es decir, se puede interpretar la radiación a partir del calor que la genera.

3.2. Antenas y radiotelescopios

Un radiotelescopio es un instrumento electrónico diseñado para captar ondas de radio provenientes de cuerpos celestes como el Sol, un planeta, el centro de la galaxia, entre otros. En general estos sistemas están conformados principalmente por tres elementos: Una antena, un receptor y un sistema para visualizar la señal. Entre los elementos del instrumento la antena reflectora es esencial para recibir o emitir ondas electromagnéticas en el medio que se encuentre. Para su uso en radiotelescopios se aprovecha su configuración como antena receptora, siendo de fácil adquisición en la actualidad gracias a la tecnología de comunicación satelital. Estos instrumentos están diseñados para captar una señal, enfocarla en el receptor (a partir de su diseño parabólico) y transmitirla a un sistema en función parámetros y características únicas para cada tipo de antena, las cuales determinan la respuesta y calidad del instrumento.

El funcionamiento de la antena se ve influenciado por la longitud de onda que estemos observando, una manera de entender el fenómeno es a través de la difracción en una rendija.

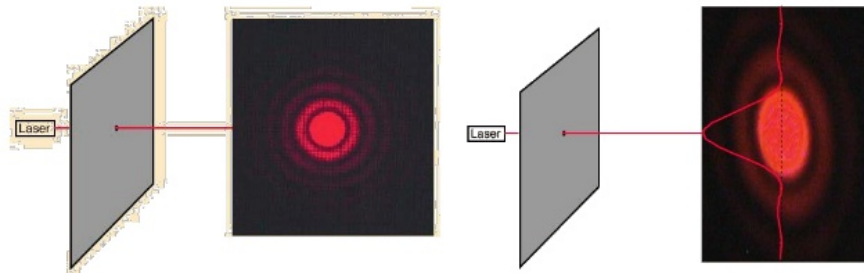


Figura A.1: Patrón de difracción por una rendija circular.

En la parte izquierda de la Figura A.1 se observa el patrón de difracción del haz que ha incidido en la rendija, donde el punto central es la luz que ha pasado directamente y las circunferencias que la rodean representan la que se difracta al pasar por los bordes de la rendija circular. Otra manera de ver este patrón se aprecia en la parte derecha de Figura A.1, donde se observan las intensidades que se han proyectado en la pantalla. Haciendo la analogía de este fenómeno al funcionamiento de antenas, podemos asumir el diámetro de la antena como el ancho de la rendija, de manera que obtendremos difracción de la luz que deseamos percibir en los bordes de esta. El patrón de las intensidades captadas por la antena se conoce como patrón de radiación.

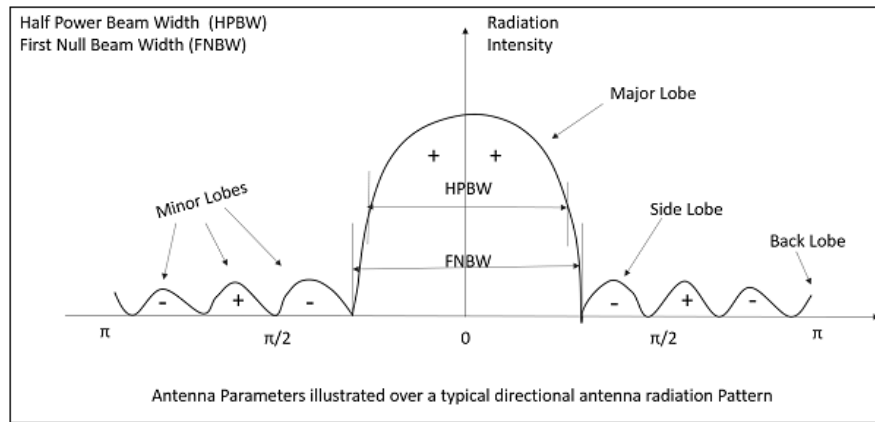


Figura A.2: Patrón de radiación típico en una antena.

Del patrón de radiación podemos obtener varias características de la antena que estamos utilizando, dos de ellas son el ancho de banda (FPBW) y la mitad de ancho de banda (HPBW) el cual se determina cuando la señal cae en 3 dB desde el punto máximo de intensidad. El significado de estos parámetros está relacionado con la capacidad de observación del instrumento, es decir, el ancho del objeto que podremos observar con ella, además del ángulo mínimo de separación entre dos objetos para poder ser identificados por la antena individualmente. Una manera de calcular la mitad del ancho de banda es a través del criterio de resolución de Rayleigh, el cual lo determina en función de la longitud de onda λ y el diámetro de la antena D . El criterio de Rayleigh da como resultado el ancho angular del haz de la antena en radianes, un valor que al ser comparado con el de telescopios ópticos deja en evidencia la poca calidad de resolución de las antenas.

$$HPBW = 1.2 \frac{\lambda}{D}$$

Una característica a tener en cuenta a la hora de cuantificar el desempeño de la antena es su eficiencia η , ya que, este parámetro representa el porcentaje del área de la antena que está

siendo iluminada y se obtiene como el cociente entre la área efectivamente iluminada y la área física.

$$\eta = \frac{A_{eff}}{A}$$

Una manera de calcular el área efectiva es a través de la ganancia de la antena como.

$$A_{eff} = \frac{G\lambda^2}{4\pi}$$

3.3. Temperatura de brillo

La temperatura de brillo o temperatura de radiación es la temperatura de un cuerpo negro en equilibrio térmico con su entorno y no es una temperatura como se entiende normalmente. Caracteriza la radiación, y dependiendo del mecanismo de radiación puede diferir considerablemente de la temperatura física de un cuerpo radiante (aunque es teóricamente posible construir un dispositivo que se caliente por una fuente de radiación con una temperatura de brillo igual a la temperatura real).

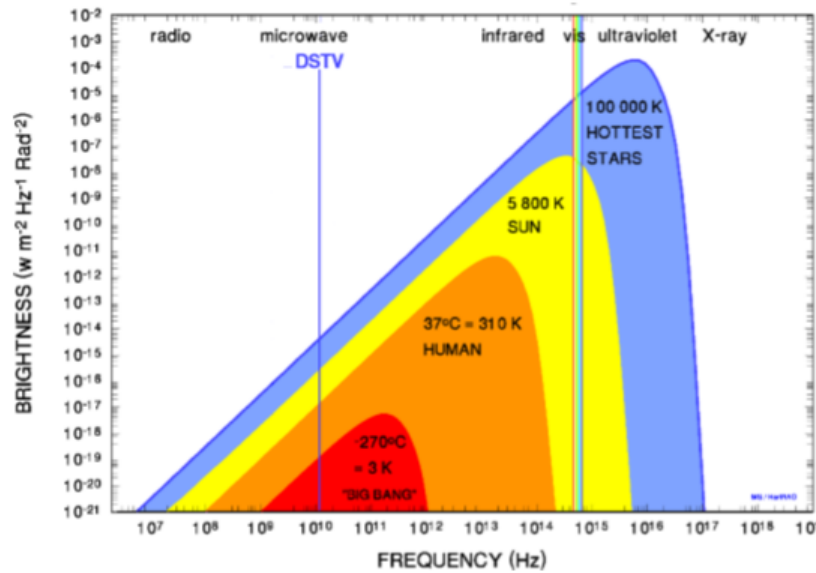


Figura A.3: Radiación de cuerpo negro de objetos con mismo tamaño angular a diferentes temperaturas.

A partir de la radiación de cuerpo negro es posible determinar la temperatura de brillo de un objeto en una frecuencia determinada. En nuestro caso los paquetes de energía de la onda son mucho menores a la energía por radiación térmica, es decir, $h \ll kT$ (donde k es la

constante de Boltzmann y h la constante de Planck), a partir de esta condición podemos aplicar la ley de Rayleigh-Jeans y definir el brillo del objeto como:

$$B = \frac{2kT}{\lambda^2}$$

La temperatura aparente de la fuente emisora a una frecuencia dada es proporcional a las ondas emitidas, dando como resultado la temperatura de brillo del objeto T_B .

3.4. Temperatura de la antena y diámetros angulares

Cuando apuntamos el radiotelescopio al objeto de interés obtendremos la lectura del cambio en la temperatura medida por la antena, este parámetro se conoce como temperatura de la antena (sin relación con la temperatura física del instrumento) T_A , inferior a la temperatura de brillo T_B .

Para determinar la temperatura medida por la antena se observa el comportamiento de esta al apuntar fuentes diferentes. Al realizar una lectura con la antena apuntando al cenit, el voltaje de respuesta entregado representará la lectura de las temperaturas de ruido del sistema (T_R) y el fondo de cielo (T_{SKY}).

$$T_R + T_{SKY} = CV_{SKY}$$

Por otro lado, al apuntar un objeto que cubra todo el ancho de haz de la antena y cuya temperatura sea conocida, será posible definir el voltaje sensado como la medición de la temperatura de referencia (T_{REF}) junto con el ruido del sistema.

$$T_R + T_{REF} = CV_{REF}$$

Al resolver el sistema planteado por las ecuaciones anteriores se cuantifica la temperatura de ruido, generado por los componentes de hardware, y la constante de proporcionalidad C entre el voltaje de respuesta y la temperatura medida por la antena. Finalmente, al apuntar el objeto de interés la lectura se puede describir de la siguiente forma.

$$T_A + T_R + T_{SKY} = CV_A$$

La temperatura de la antena T_A se debe corregir teniendo en cuenta el ancho angular del objeto y el ancho angular del haz de la antena.

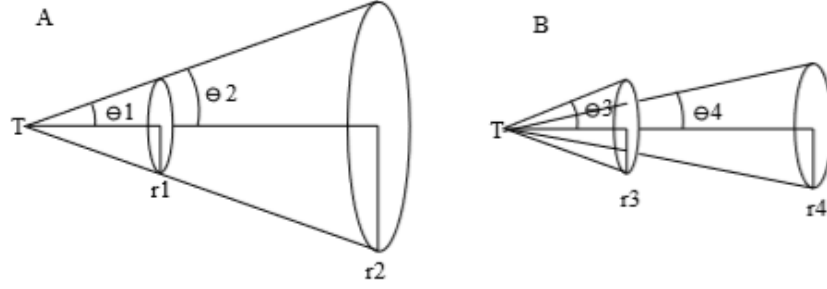


Figura A.4: Comparación del ancho angular con el ancho físico.

El problema generado por la diferencia entre los anchos angulares se explica en la Figura A.4. En el diagrama A se define una situación ideal donde el radio físico de los objetos es diferente ($r_1 < r_2$), pero el ancho angular al observarlos desde el punto T es el mismo ($\theta_1 = \theta_2$). En el diagrama B la situación cambia, ya que, el radio físico de los objetos es el mismo ($r_3 = r_4$), pero los anchos angulares son diferentes para el observador en T, una representación que se asemeja a la situación actual al realizar las mediciones con el radiotelescopio. Una manera determinar mejor la zona ocupada por un objeto pequeño, en un caso de dos dimensiones, es a través de su ángulo sólido (área angular), un parámetro que depende del ancho angular del objeto θ y se define como Ω en estereorradianes.

$$\Omega = \pi(1 - \cos\theta)$$

Para nuestro caso el ángulo θ es menor a 10° , lo que nos permite reescribir el coseno reduciendo la expresión para el ángulo sólido.

$$\Omega = \pi\theta^2$$

Teniendo esto en cuenta se hace la corrección en la temperatura medida por la antena, dando como resultado la verdadera temperatura de brillo del objeto observado.

$$T_B = \left(\frac{\Omega_A}{\Omega_B} \right) T_A$$

donde

- Ω_A = Área angular del haz completo de la antena ($\theta = 2 * HPBW$).
- Ω_B = Área angular del objeto observado ($\theta = \theta_{sol}$).
- T_A = Temperatura observada por la antena.

4. Materiales

4.1. Antena de plato reflector

La antena está elaborada a partir de un disco de 1.2 m de diámetro (con una eficiencia de 0.57), en malla cuadriculada de 2.8 mm, además, la distancia focal del diseño es de 0.72 m.



Figura A.5: Antena de plato reflector.

4.2. Controlador de la montura

Se cuenta con el controlador SPID MD-02, el cual permite realizar el protocolo de los motores configurados en azimuth y altitud simultáneamente.



Figura A.6: Controlador SPID MD-02.

La Figura A.6 tiene una vista frontal del instrumento y cada una de sus partes:

1. Teclado.
2. Display de 20x20.
3. Botón encendido-apagado.

4.3. Buscador de satélites y filtro pasa bajos

La señal obtenida es filtrada para eliminar el ruido de altas frecuencias, por lo que La señal proveniente de la antena será llevada al buscador de satélites, el cual indicará la intensidad de la lectura a través de la aguja movida por el galvanómetro interno del instrumento.

4.4. Adquisición y procesamiento de datos

Una vez se ha condicionado la señal se pasa a la digitalización de esta. Se lleva entonces la salida del filtro a una tarjeta de adquisición de datos (DAQ) para luego procesar la información a través de un ordenador implementando una Interfaz Gráfica de Usuario elaborada en Python.

5. Procedimiento

Una vez se ha realizado la lectura completa de la guía, se procede a preparar el instrumento para su operación. Esta etapa consiste en encender la fuente SPID PS-02 y esperar unos segundos mientras encienden los LEDs indicadores, una vez encendidos se procede a encender el controlador SPID MD-02. En este punto los motores de la antena están energizados y se pueden manipular directamente desde los botones del controlador. Como la operación durante la práctica se realizará a través del Software CRT se procede a conectar al PC el cable USB que sale por detrás del controlador y se ubica junto al cable coaxial que trae la señal del LNB, este nos permitirá la comunicación entre el controlador y el PC.

Ahora se continúa a realizar las conexiones necesarias para la adquisición de datos, para ello se debe tener a la mano el Buscador de Satélites, la DAQ y la Caja de Filtrado y Alimentación. Primero se prepara la DAQ para la comunicación con el PC, para ello se conecta el cable de datos entre el PC y la DAQ. Seguidamente se hacen las conexiones en la entradas analógicas de la DAQ para recibir la señal procedente del instrumento. Aquí es necesario puentear los pines AGND y 0- de la DAQ haciendo uso de algún cable corto, donde allí mismo se introducirá la tierra (cable negro con líneas blancas) de la salida de la caja de filtrado y alimentación llamada “Salida de señal”. Al pin 0+ se conecta el cable restante

(Cable totalmente negro). Se procede a conectar el cable coaxial que sale por el extremo derecho del buscador de satélites a la entrada coaxial que tiene la caja de alimentación y filtrado llamada “Alimentación LNB y BS”, a su vez se conecta el cable que sale por la parte posterior del buscador de satélites al socket de entrada de la caja marcada como “Entrada de Señal”. Luego es necesario conectar al buscador de satélites por su extremo izquierdo el cable coaxial que cuelga y se encuentra enrollado debajo de la fuente PS-02 y controlador de la montura MD-02, este cable trae la señal desde el LNB. Finalmente se revisan las conexiones anteriores y se procede a conectar la caja de alimentación y filtrado a un toma de corriente, se espera unos segundo mientras el LED indicador se enciende y se presiona el interruptor para energizar todo el sistema anteriormente ensamblado. Para finalizar abrimos el Software PstRotator y seguidamente el Software CRT. En este punto el instrumento se encuentra listo para operar y presentar datos en pantalla.

Ahora, para realizar el cálculo de la temperatura media de brillo del sol, es necesario apuntar la antena reflectora a tres zonas diferentes, de esta manera se podrá observar el comportamiento de la antena ante diferentes excitaciones. Antes de iniciar lleve la salida del cable coaxial al buscador de satélites, ajuste la aguja del galvanómetro en uno y espere que este se estabilice, de esta forma se define el punto de referencia.

- **Toma del cielo de fondo:** Al iniciar la GUI, se indica como primer paso la toma de la respuesta en voltaje de la antena al apuntar al cenit, en este punto la temperatura de fondo es de 20 K aproximadamente. Ponga las coordenadas (0, 90) en el software de control de la antena y realice la toma de datos.
- **Toma de una temperatura de referencia:** Una vez se ha realizado la adquisición del fondo, se continúa con la adquisición de la señal al apuntar hacia una superficie cuya temperatura sea conocida. En nuestro caso se toma como referencia la cúpula del telescopio en el OAUTP, esta es captada direccionando el radiotelescopio en el punto (224, 0). Una vez apuntada la antena realice la toma de datos y mida la temperatura superficial de la cúpula con el pirómetro.
- **Toma de la iluminación del sol:** Finalmente se obtienen las emisiones provenientes del sol. En este paso la toma de datos conlleva un tiempo de 30 minutos, ya que, se observará un tránsito solar. Para la adquisición apunte la antena a las coordenadas acimutales del Sol adelantadas 15 minutos a través del software de control e inicie la adquisición de datos.

Finalizada la sesión se podrá observar en pantalla cada uno de los valores del voltajes adquiridos durante el desarrollo. La aplicación permite visualizar a la salida la temperatura de brillo, realizando los cálculos con el ancho de haz obtenido directamente del patrón de radiación.

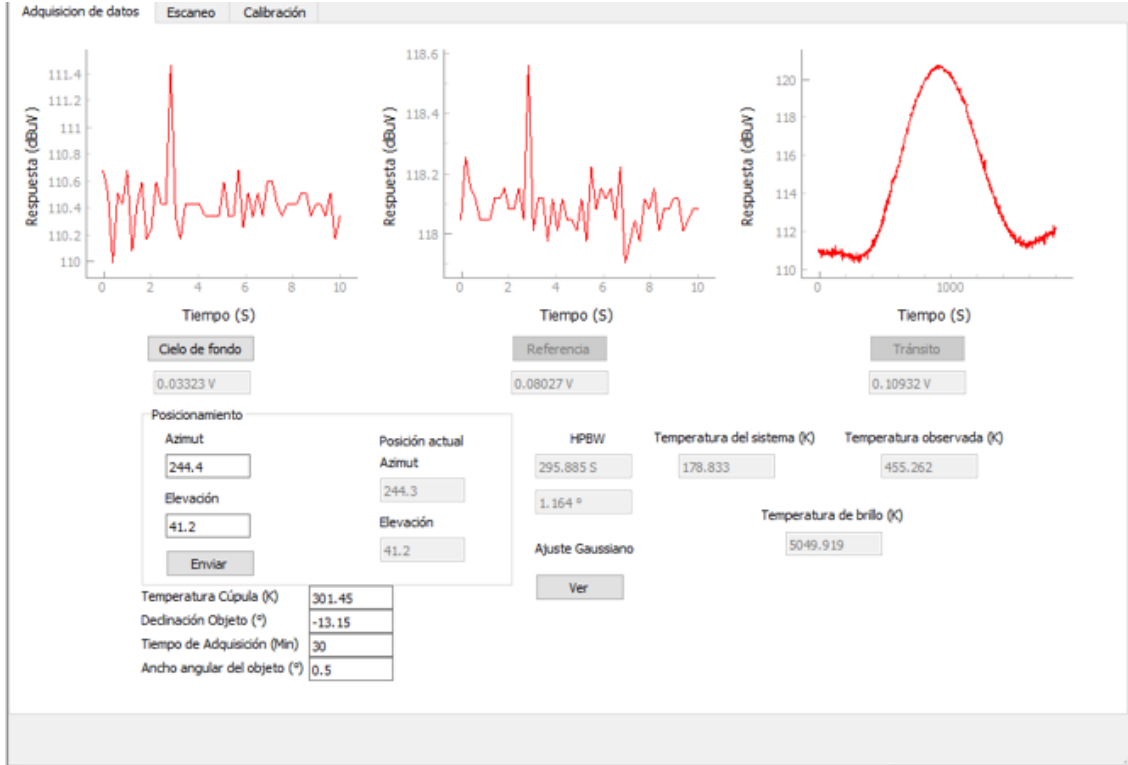


Figura A.7: Resultado de la adquisición de datos.

El programa entrega un valor aproximado para la mitad de ancho de haz de la antena en segundos, para obtener su valor en grados se tiene en cuenta que el sol se mueve a 1° cada 4 minutos a través de la bóveda celeste cuando la eclíptica está sobre la línea del ecuador. Esta velocidad aparente varía dependiendo de la declinación del sol, por lo que la conversión a grados se define como:

$$HPBW[^\circ] = \frac{1^\circ}{240S} \cdot HPBW[S] \cdot \cos(declinacionsolar[^\circ])$$

Además, el sol observado desde la tierra tiene un ancho angular de 0.508 grados. La declinación solar puede ser extraída de la página de Starpath¹.

¹https://www.starpath.com/blog_files/Table%20of%20the%20Declination%20of%20the%20Sun.pdf

6. Procesamiento de datos

Calcule el HPBW teórico de la antena y compárelo con el valor obtenido experimentalmente. En caso de encontrar variaciones considerables en el valor ¿Cuál sería la razón de dichos cambios?

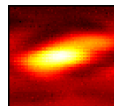
Halle el cociente entre el área angular del haz de la antena y el área angular del objeto observado. ¿Qué significado tiene esta magnitud?

Con los datos experimentales hallados en el numeral 5, obtenga la temperatura de la antena y la temperatura media de brillo del Sol.

Anexo B

Interfaz Gráfica de Usuario

B.1 Guía de usuario



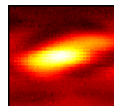
Compact Radio Telescope (CRT)

Software para la adquisición y procesamiento de datos de radiotelescopios compactos

Versión 1.0

Guia de usuario

2020



Contenido

1. Consideraciones generales

- 1.1 ¿Qué es *Compact Radio Telescope*?
- 1.2. Contenido del CD
- 1.3. Licencia y Términos de Uso
- 1.4. ¿Cómo adquirir *Compact Radio Telescope*?

2. Guía de instalación

- 2.1. Requerimientos mínimos del sistema
- 2.2. Instalación

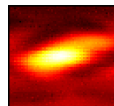
3. Presentación de la interfaz gráfica de usuario

- 3.1. Ventana Adquisición de datos
- 3.2. Ventana Escaneo
- 3.3. Ventana Calibración
- 3.4. Menú de ayuda
 - 3.4.1. Acerca de
 - 3.4.2. Guía de usuario

4. Ejemplos de aplicación

5. Algoritmos

- 5.1. Comunicación DAQ
- 5.2. Envío y recepción de coordenadas
- 5.3. Ajuste Gaussiano
- 5.4. Ventana Adquisición de datos
- 5.5. Ventana Escaneo
- 5.6. Ventana Calibración



1. Consideraciones generales

1.1. ¿Qué es *Compact Radio Telescope*?

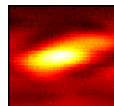
Compact Radio Telescope es una aplicación desarrollada en Python y QtDesigner que permite estimar las características de un radiotelescopio compacto a partir de la observación de cuerpos celestes a través de diferentes técnicas. Es así como la aplicación se divide en tres ventanas, cada una de estas enfocada a un método diferente de observación. De forma predeterminada se despliega la ventana llamada “Adquisición de datos”. En esta se facilita la calibración del sistema de recepción, además de la observación de cuerpos celestes a través de un tránsito, dando como resultado el ancho de haz de la antena, la temperatura del sistema y la temperatura de brillo observada. Por otro lado, la ventana “Escaneo” permite realizar la observación de una zona predefinida de la bóveda celeste al mover una de las direcciones (acimut o altitud) de forma oscilatoria durante un tiempo establecido. Esta ventana da como resultado una imagen mapeada y en escala de grises de la zona observada en contraste con la intensidad recibida. Finalmente, la ventana “Calibración” permite ejecutar la observaciones en un rango específico de acimut y altitud, dando como resultado la representación gráfica de la intensidad recibida por el radiotelescopio al observar cada posición dentro del rango especificado. Además, arroja el error presente en el apuntado de la antena, permitiendo caracterizar su magnitud durante cada sesión y facilitando su recopilación durante distintas sesiones.

Este documento es una guía de usuario que le permitirá instalar la aplicación y comprender su funcionamiento.

1.2. Contenido del CD

El CD de instalación de *Compact Radio Telescope* contiene los siguientes archivos:

- CRT.exe: Instalable de la aplicación.
- Guia.pdf: Este archivo de ayuda.
- Readme.txt: Archivo de texto con instrucciones para la instalación y ejecución del programa.
- Acercade.png: Imagen de “Acerca de” la aplicación.
- Logoapp.ico: Ícono de la aplicación.



1.3. Licencia y Términos de Uso

Compact Radio Telescope es una aplicación desarrollada por el Grupo de Investigación en Astronomía Alfa Orión del Observatorio Astronómico de la Universidad Tecnológica de Pereira, de uso libre (free software). Se solicita a los usuarios que, en caso de generarse publicaciones a partir de observaciones logradas haciendo uso de este software, se realicen las citaciones respectivas.

1.4. ¿Cómo adquirir *Compact Radio Telescope*?

Compact Radio Telescope está disponible para su descarga en la siguiente dirección Web:
<http://observatorioastronomico.utp.edu.co>

2. Guía de instalación

2.1. Requerimientos Mínimos del Sistema

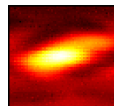
Para el funcionamiento apropiado de la aplicación *Compact Radio Telescope*, se recomienda que el sistema cumpla, como mínimo, con los siguientes requerimientos:

- Sistema operativo Microsoft Windows®7 de 64bits.
- Procesador de 1.4GHz.
- Memoria RAM de 1 GB.
- 300 MB de espacio libre en el disco duro.

2.2. Instalación

Para la instalación de *Compact Radio Telescope* siga los siguientes pasos:

- 1 Instale los controladores de la tarjeta de adquisición de datos: *LabView*, *Multsim* y *NI Elvis*.
- 2 Instale el software para el control de la montura *Pstrotator*.
- 3 Configure el tipo de motor usado para el movimiento en acimut en *Pstrotator* a través de la siguiente ruta: Setup → AZ Controller (para nuestro caso es SPID BIG-RAS/HR 0.1deg). Ya que la montura debe ser altacimutal, también debe ser configurado el motor de elevación en la siguiente ruta: Setup → EL/AZ+EL Controller (para nuestro caso es SPID RAS/HR 0.2deg).



- 4 Seleccione el protocolo de control para la montura seleccionada a través de: Setup → Controllers Setup (en nuestra montra es SPID RAS/HR 0.2deg a 460800 baud rate) .
- 5 Configure la velocidad de lectura de los motores en Setup → Rotor Refresh Rate y seleccione la lectura cada 0.5 s.
- 6 Seleccione el puerto COM de comunicación de la montura con Pstrotator en la ruta Communication → EL/AZ+EL COM Port.
- 7 Habilite la comunicación UDP de *Pstrotator* en “Setup”.
- 8 En la ruta Communicaction → UDP Controll port, establezca el puerto 12000 como predeterminado y active el reporte de posición automático mientras los motores se encuentren en movimiento.
- 9 Ejecute el instalable CRT.exe del CD.
- 10 Conceda permisos de acceso a redes publicas y privadas a *Pstrotator* y *Compact Radio Telescope*.
- 11 Ejecute el archivo CRT.exe resultante de la instalación disfrute y la aplicación.

Una vez finalizada la instalación se creará la carpeta “CRT” en documentos, donde se almacenarán todos los resultados de las sesiones de adquisición realizadas.

3. Presentación de la interfaz gráfica de usuario

La aplicación *Compact Radio Telescope* fue construida implementando la herramienta QtDesigner, la cual permite crear rápidamente interfaces gráficas de usuario con widgets del marco Qt GUI. Por otro lado, las utilidades, entradas y salidas de texto, se logran a través de enlaces manipulados con PyQt5. De esta forma se une el conjunto de bibliotecas de la herramienta Qt, perteneciente a C++, con las bibliotecas de Python, logrando una interfaz cómoda y amigable con el usuario. Presentamos una aplicación que permite realizar el movimiento en acimut y altitud de radiotelescopios compactos, cuya montura pueda ser controlada a través de Pstrotator, además de realizar la adquisición de datos del radorreceptor simultáneamente. Con estas utilidades, las funciones de la aplicación se divide en tres ventanas diferentes como se muestra en el diagra de bloques de la Figura B.1.

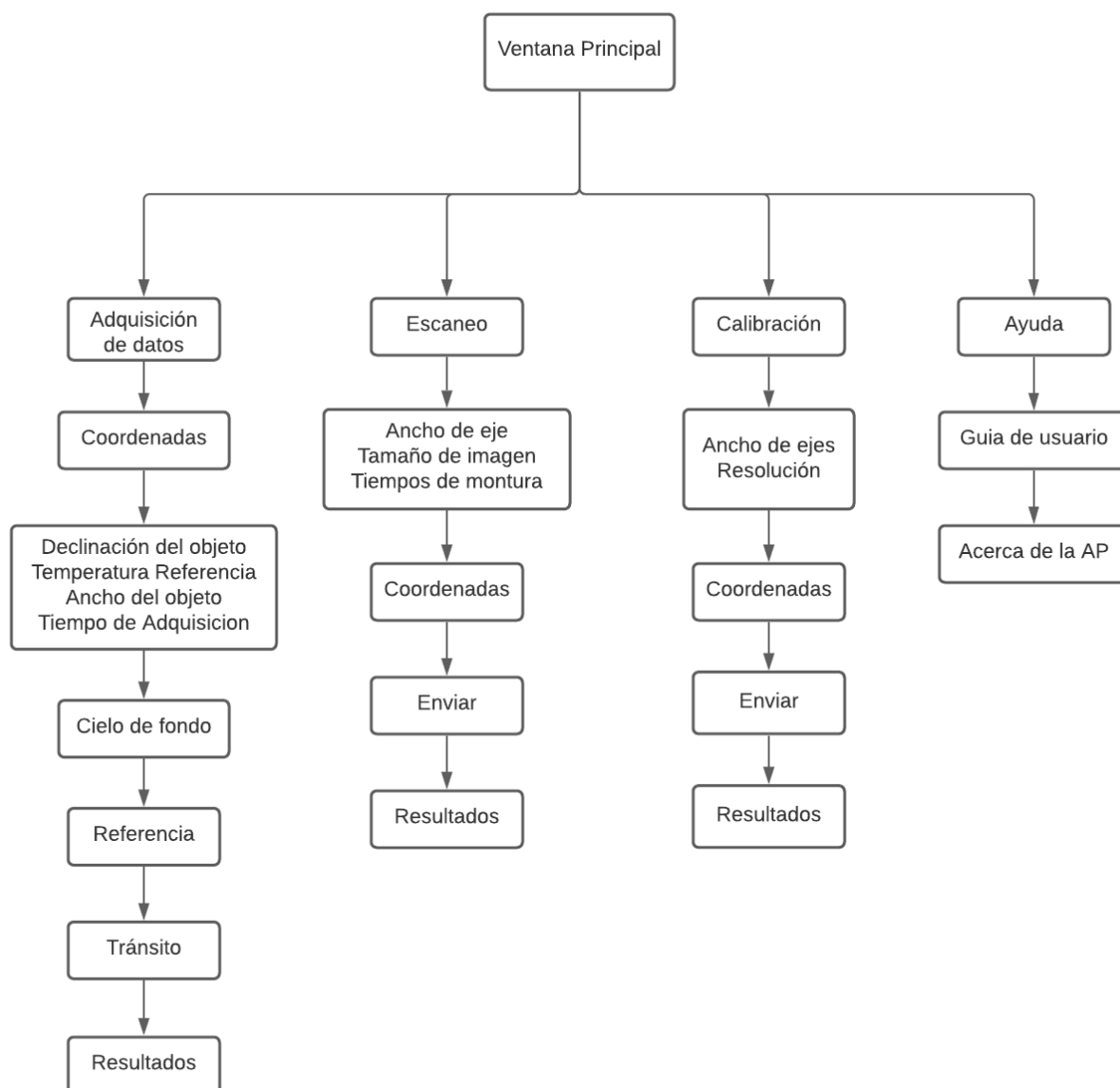
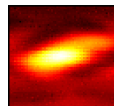


Figura B.1: Estructura general *Compact Radio Telescope*

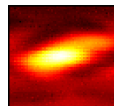
3.1. Ventana Adquisición de Datos

Una vez ejecutada la aplicación, se visualizará de forma predeterminada la pestaña “Adquisición de datos”, mostrada en la Figura B.2. Esta se centra en guiar la sesión de adquisición en tres etapas, bajo una secuencia basada en el procedimiento involucrado en el cálculo de la temperatura de brillo de un cuerpo celeste.



Figura B.2: Pestaña Adquisición de Datos.

Para iniciar se debe enviar el radiotelescopio hacia una zona del cielo donde se observe el espacio vacío. Para esto ingrese las coordenadas donde desea realizar la adquisición través del cuadro “Posicionamiento” como se muestra en la Figura B.3. Si las coordenadas ingresadas no se encuentran en el rango indicado (acimut de 0° a 360° y altitud de 0° a 90°), o tienen número decimal impar, se desplegará la advertencia de la Figura B.4, evitando el envío de las coordenadas para que estas puedan ser ingresadas correctamente.



Adquisición de datos Escaneo Calibración

Cielo de fondo

Referencia

Tránsito

Posicionamiento

Azimut:

Elevación:

Posición actual

Azimut:

Elevación:

HPBW:

Temperatura del sistema (K):

Temperatura observada (K):

Temperatura de brillo (K):

Ajuste Gaussiano:

Temperatura Cúpula (K):

Deducción Objeto (°):

Tiempo de Adquisición (Min):

Ancho angular del objeto (°):

Figura B.3: Ingreso de coordenadas.

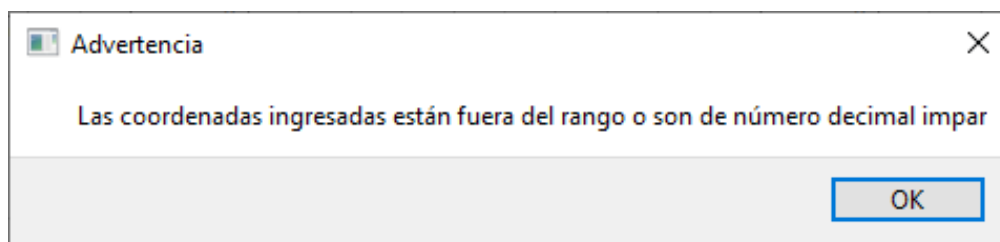


Figura B.4: Mensaje de advertencia para coordenadas erróneas.

Una vez finalizado el movimiento presione el botón “Cielo de Fondo”, obteniendo la respuesta del radioreceptor ante el cielo frío como se observa en la Figura B.5.

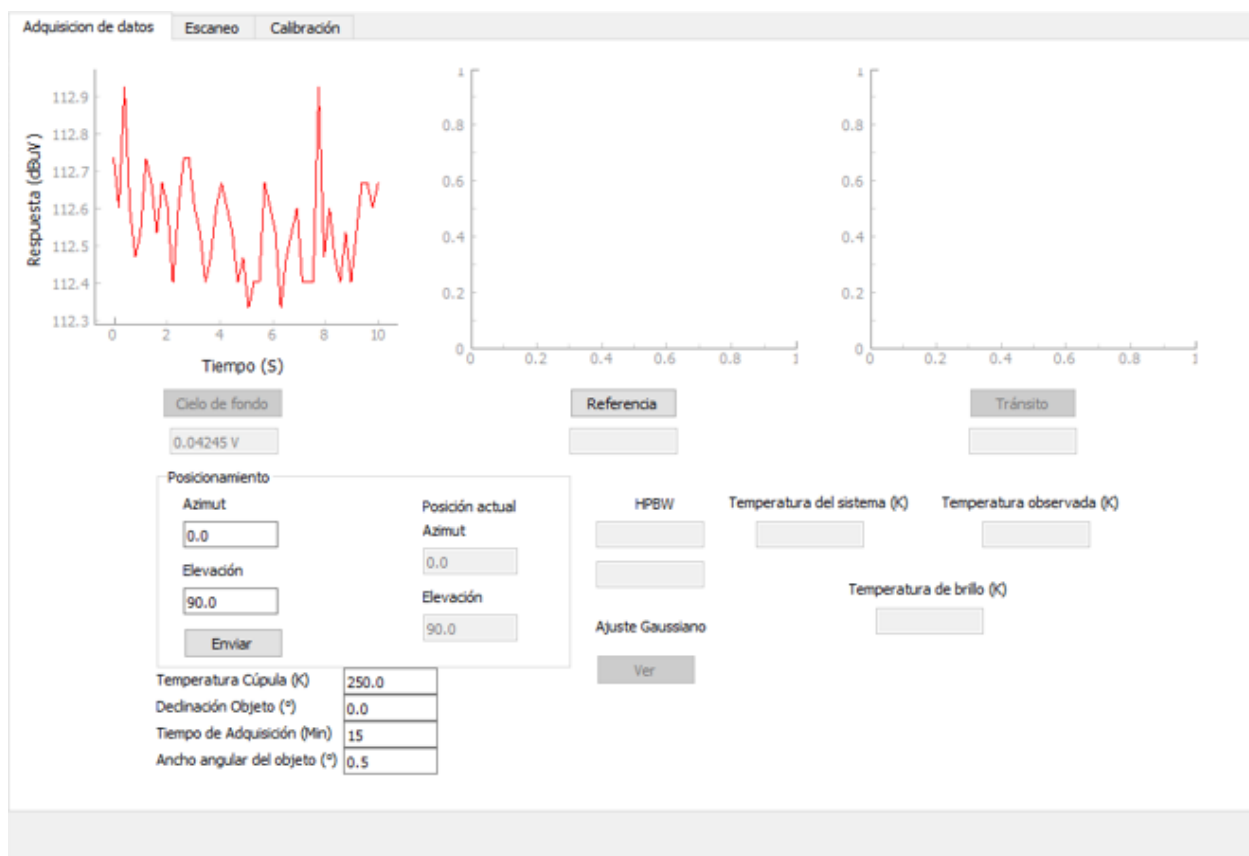
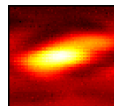


Figura B.5: Resultado al oprimir “Cielo de fondo”.

Una vez determinada el voltaje promedio de la respuesta recibida al apuntar el espacio vacío, para el cual se asume una temperatura teórica de 20 K, se habilita el botón “Referencia”. En esta etapa se debe hacer la adquisición al apuntar un objeto que ocupe todo el ancho de haz de la antena y cuya temperatura sea conocida. Para nuestro caso, se envía el radiotelescopio al punto 224 ° en acimut y 0 ° en altitud, lugar donde se encuentra la cúpula del OAUTP. Una vez finalizado el movimiento se oprime el botón “Referencia”, dando el resultado mostrado en la Figura B.6.

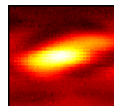


Figura B.6: Resultado al oprimir “Referencia”.

Como etapa final, se habilita el botón “Tránsito”. Antes de realizar la adquisición se debe tener en cuenta los parámetros descritos en la zona inferior de la pestaña. Aunque sus valores se encuentran de forma predeterminada, estos son cruciales en el resultado del proceso. El primero de estos es la “Temperatura de Referencia”, la cual debe ser ingresada al medir la temperatura del objeto apuntado durante la segunda adquisición de datos. El segundo es “Declinación del objeto”, la cual debe ser ingresada en grados decimales. Finalmente, se debe ingresar el parámetro “Tiempo de adquisición”, el cual depende de la duración del tránsito captar y debe ser mayor a 1 min. El resultado final al ingresar los parámetros se observa en la zona inferior de la Figura B.6. Con el fin de alcanzar mejores resultados se recomienda realizar un tránsito completo. Para esto se debe enviar el radiotelescopio hacia un punto donde se encontrará el objeto en un tiempo de 15 minutos. Una vez finalizado el movimiento, se oprime el botón “Tránsito”, el cual realiza la adquisición dejando que el objeto entre y salga del haz de la antena, caracterizando su patrón de radiación. Mientras el proceso está en curso se despliega el mensaje mostrado en la Figura B.7, ya que, durante la adquisición, el programa no responderá.

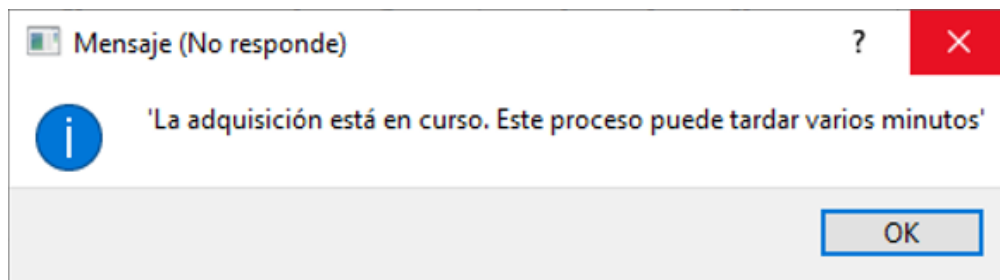


Figura B.7: Mensaje de espera mientras se ejecuta la adquisición.

Los resultados arrojados por la ventana serán el ancho de haz de la antena, en grados y segundos, la temperatura del sistema, la temperatura observada por radiotelescopio y la temperatura de brillo del objeto apuntado como lo visualiza la Figura B.8.

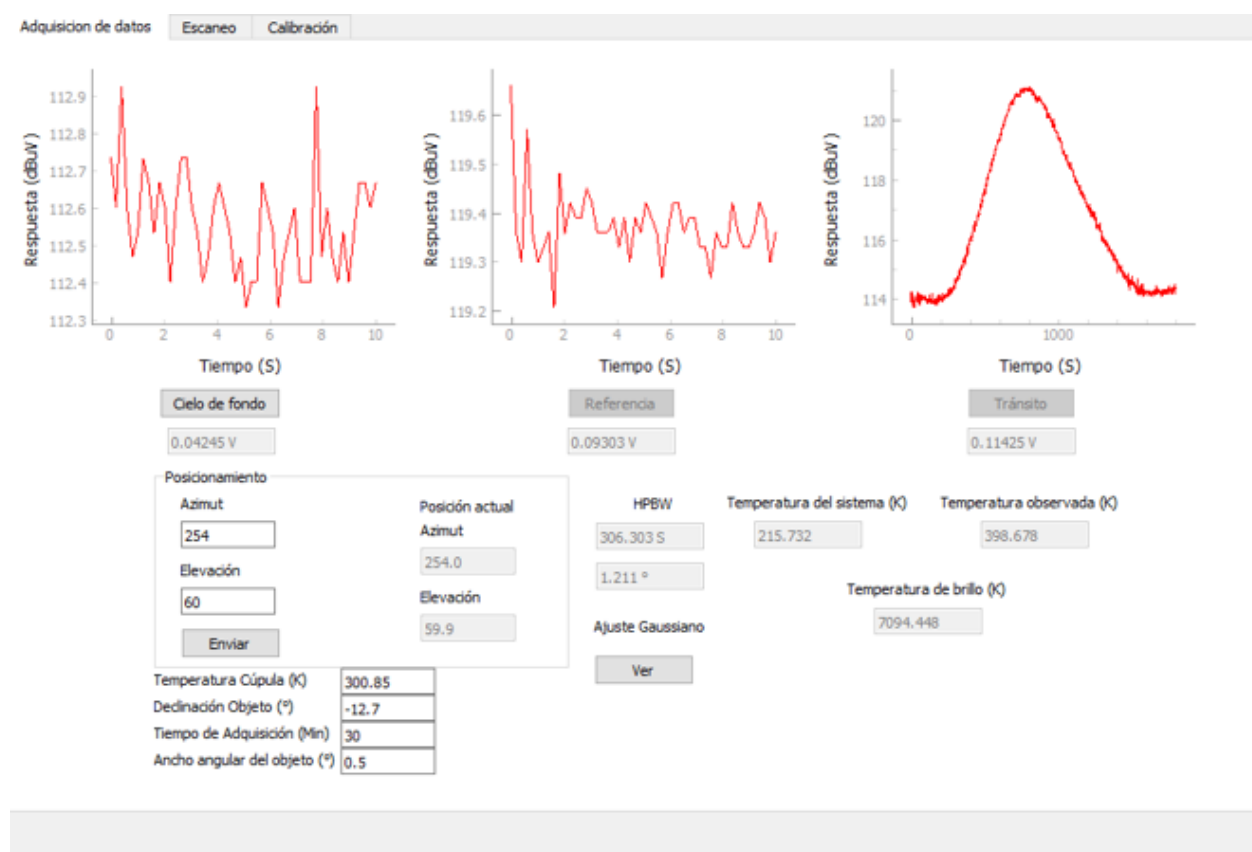


Figura B.8: Resultado final.

Además, al finalizar la adquisición se habilita el botón “Ver”, como se observa en la Figura B.9. Este nos permite visualizar el ajuste Gaussiano más aproximado patrón de radiación de la antena resultante del tránsito.

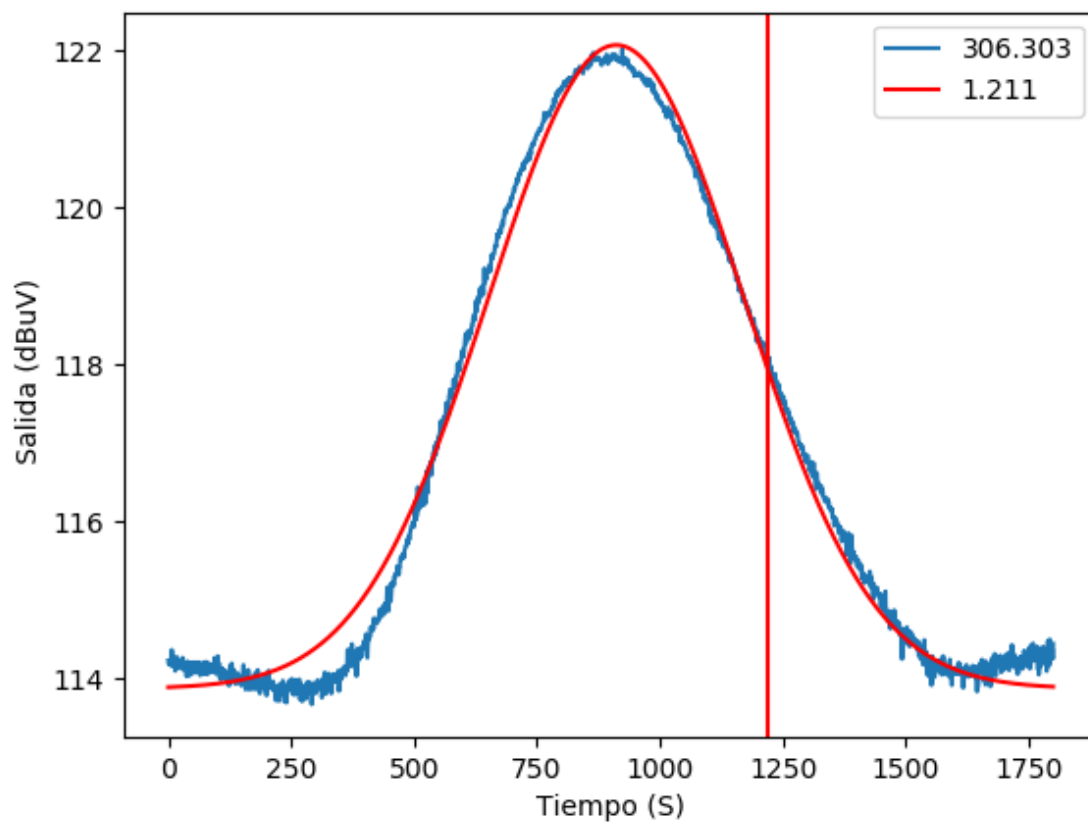
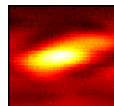


Figura B.9: Ajuste Gaussiano durante la sesión.

3.2. Ventana Escaneo

Al ingresar en la pestaña “Escaneo”, habilitaremos la segunda función de la aplicación, la cual facilita la ejecución de escaneos de deriva en una zona específica del cielo. Antes de iniciar la adquisición es necesario configurar el tiempo que le toma al telescopio moverse en cada una de las coordenadas. Ya que la ventana es realizable con un ancho de ejes de 2° , 4° y 6° es necesario caracterizar el tiempo que le toma a la montura moverse en estas magnitudes. Como se muestra en la Figura B.10, ingresa la coordenada central de la ventana. Si el punto coordinado ingresado no se encuentra en el rango definido para acimut y altitud, o no contiene número decimal par, se desplegará la advertencia de la Figura B.4

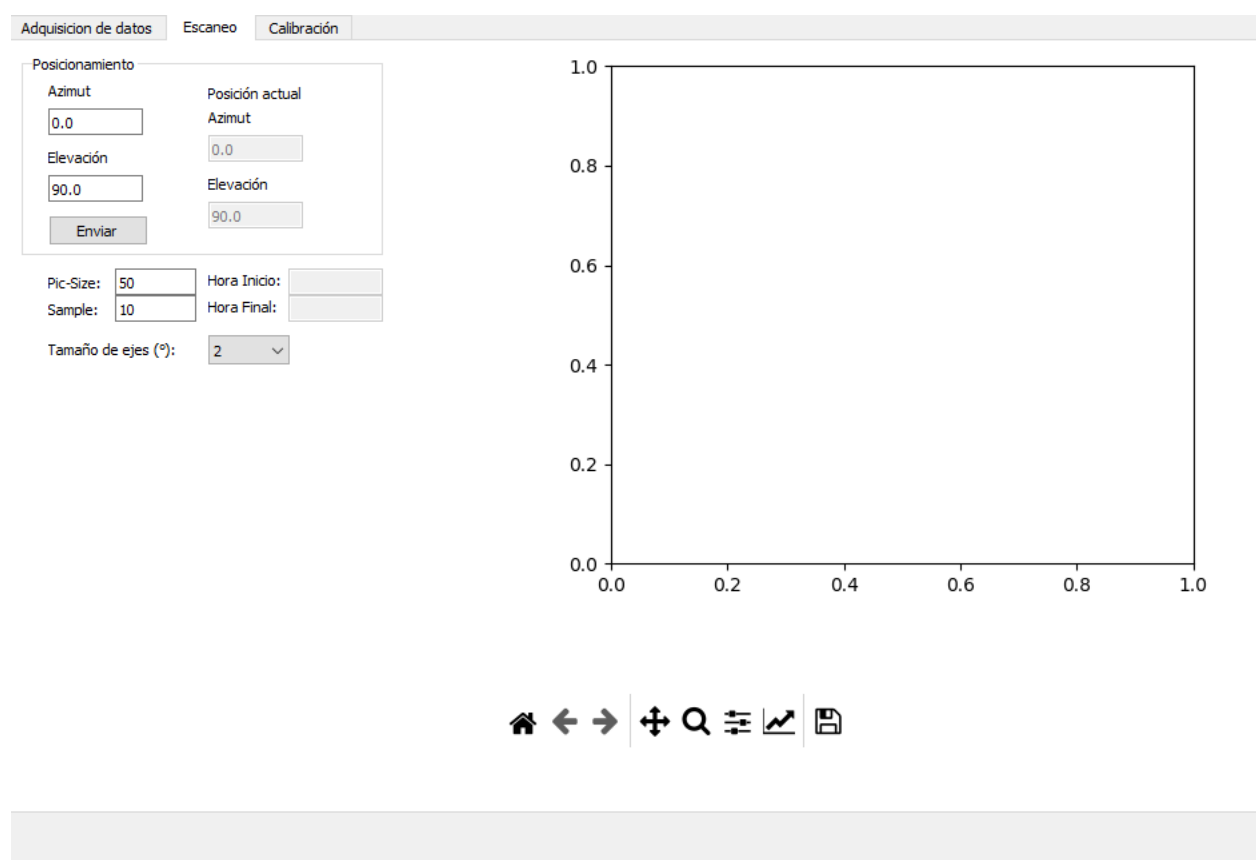


Figura B.10: Ventana Escaneo.

Ahora, es necesario escoger el ancho de la ventana y el tamaño de la imagen resultante. Para elegir el ancho de la ventana se debe dar click en “Ancho de los ejes”, donde se desplegarán las opciones disponibles. En nuestro caso seleccionaremos un ancho de 6° . A continuación, para definir el tamaño de la imagen, se debe ingresar los parámetros “Pic-Size” y “Sample Rate”, los cuales definen la cantidad de veces que se repetirá el movimiento y el número de muestras tomadas en cada oscilación. En el caso de “Sample Rate”, no se pueden superar las

300 muestras por segundo. Para este ejemplo, el movimiento se realizó 700 veces, con una velocidad de muestreo de 300 datos por segundo, como se observa en la Figura B.11. Una vez se ha iniciado la adquisición se despliega el mensaje de espera de la Figura B.7

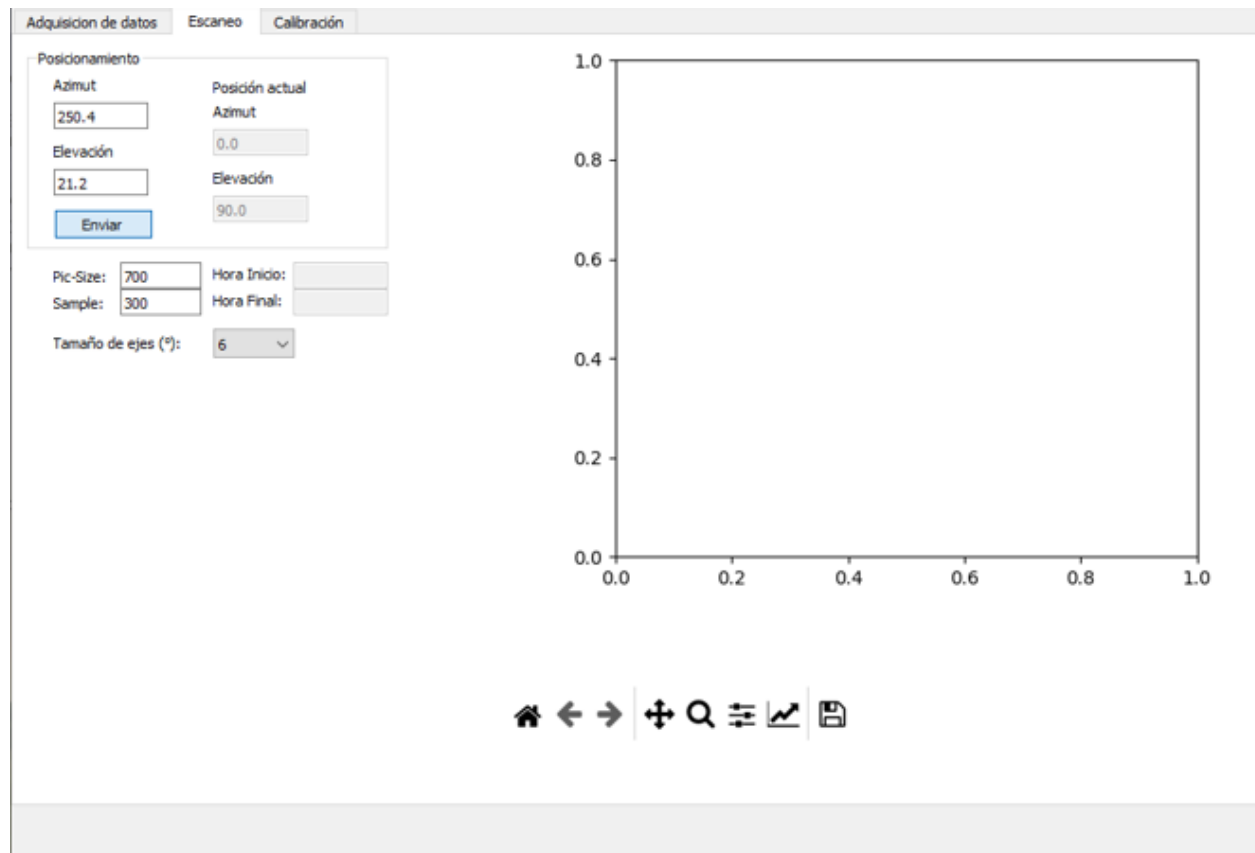


Figura B.11: Ingreso de los parámetros.

Finalmente, se debe ingresar la coordenada central de la ventana en la caja “Posicionamiento” como se muestra en la zona superior de la Figura B.11. Al oprimir el botón enviar, el radiotelescopio se moverá de forma oscilatoria en la coordenada de acimut o altitud, dependiendo de la posición del objeto, con la amplitud escogida, mientras la otra coordenada se mantiene en una posición constante. Como resultado se observará en la zona derecha la ventana la imagen del mapeo en escala de color, mientras que, en la zona inferior izquierda se observa la imagen en blanco y negro (ver Figura B.12).

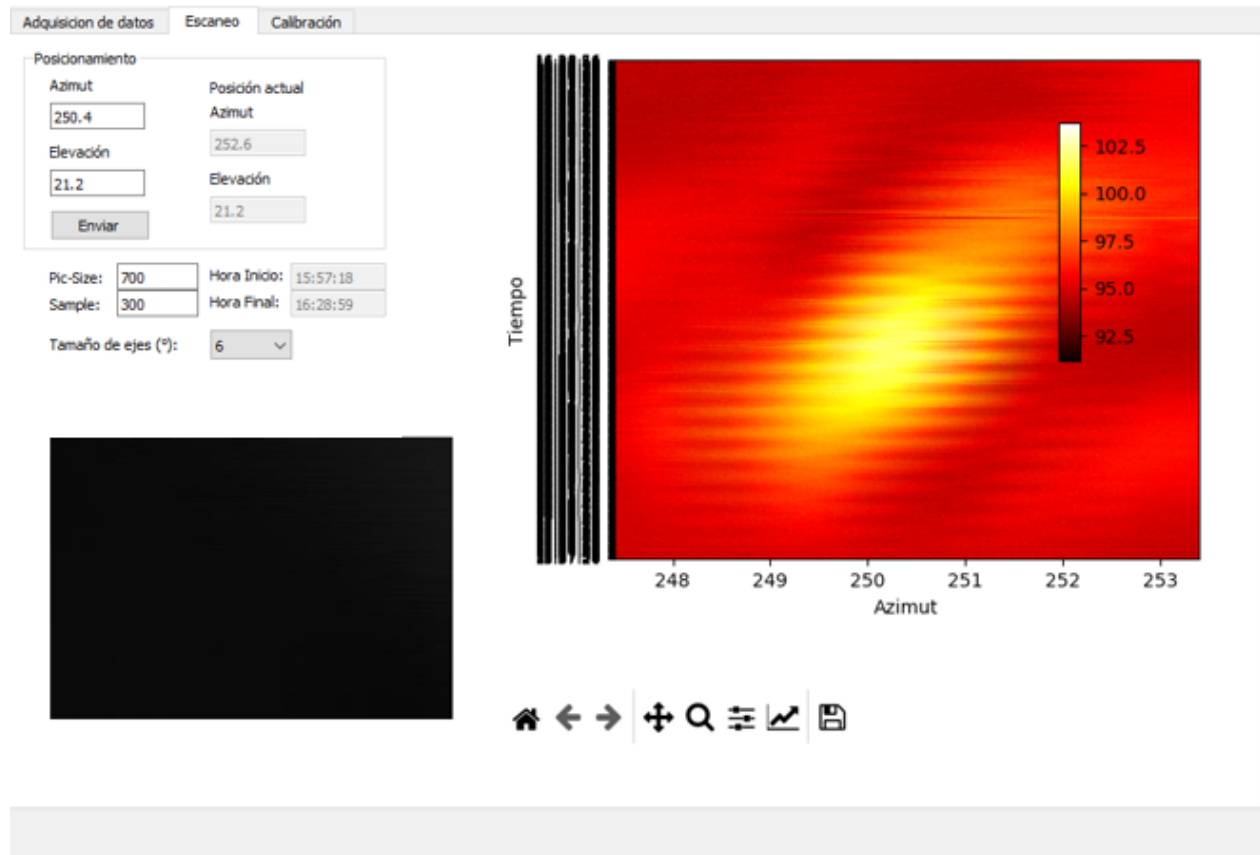
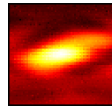


Figura B.12: Resultado final.

3.3. Ventana Calibración

Al ingresar en la ventana “Calibración”, se despliega la pestaña mostrada en la Figura B.13, la cual permite realizar la observación del cielo a través de una matriz cuadrada de tamaño predefinible. En esta cada punto representa un valor de intensidad medido, según la posición en la que se encuentra apuntado el radiotelescopio.

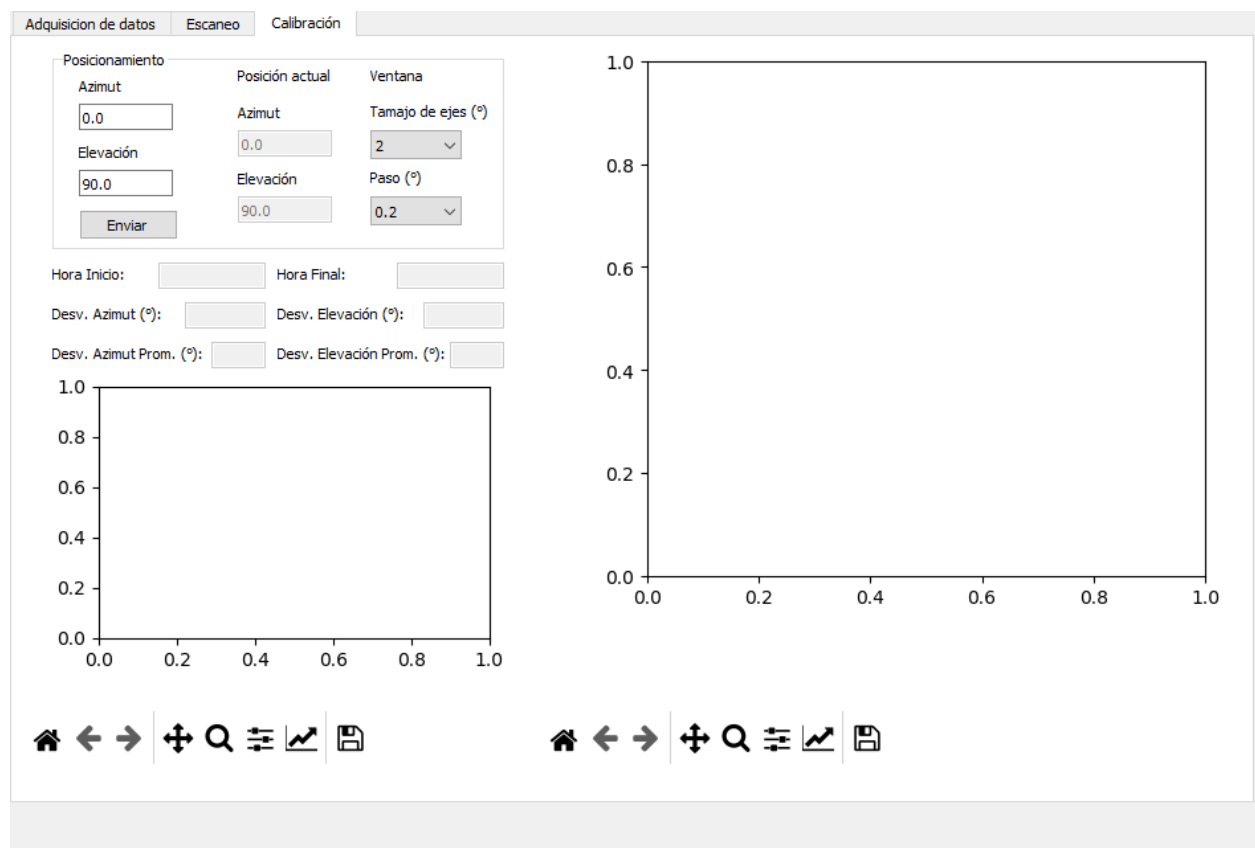


Figura B.13: Ventana Calibración.

Antes de realizar el proceso de adquisición es necesario definir el tamaño de la ventana en el desplegable “Ancho de ejes” y la resolución del movimiento en el desplegable “Paso” junto con la coordenada central. Si el punto coordenado ingresado no se encuentra en el rango definido para acimut y altitud, o no contiene número decimal par, se desplegará la advertencia de la Figura B.4. Para nuestro ejemplo utilizaremos una matriz con un ancho de $4^\circ \times 4^\circ$, recorrida a un paso de 0.4° . Una vez se han ingresado los parámetros para definir el tamaño de la matriz, se debe enviar el radiotelescopio a la posición donde se encuentra el objeto de interés. Para nuestro ejemplo utilizaremos el satélite geostacionario IntelSat 30 ubicado a 256.2° en acimut y 66.4° en altitud desde nuestro lugar de observación. Este proceso se ilustra en la Figura B.14. Una vez se ha iniciado la adquisición se despliega el mensaje de espera de la

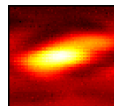


Figura B.7

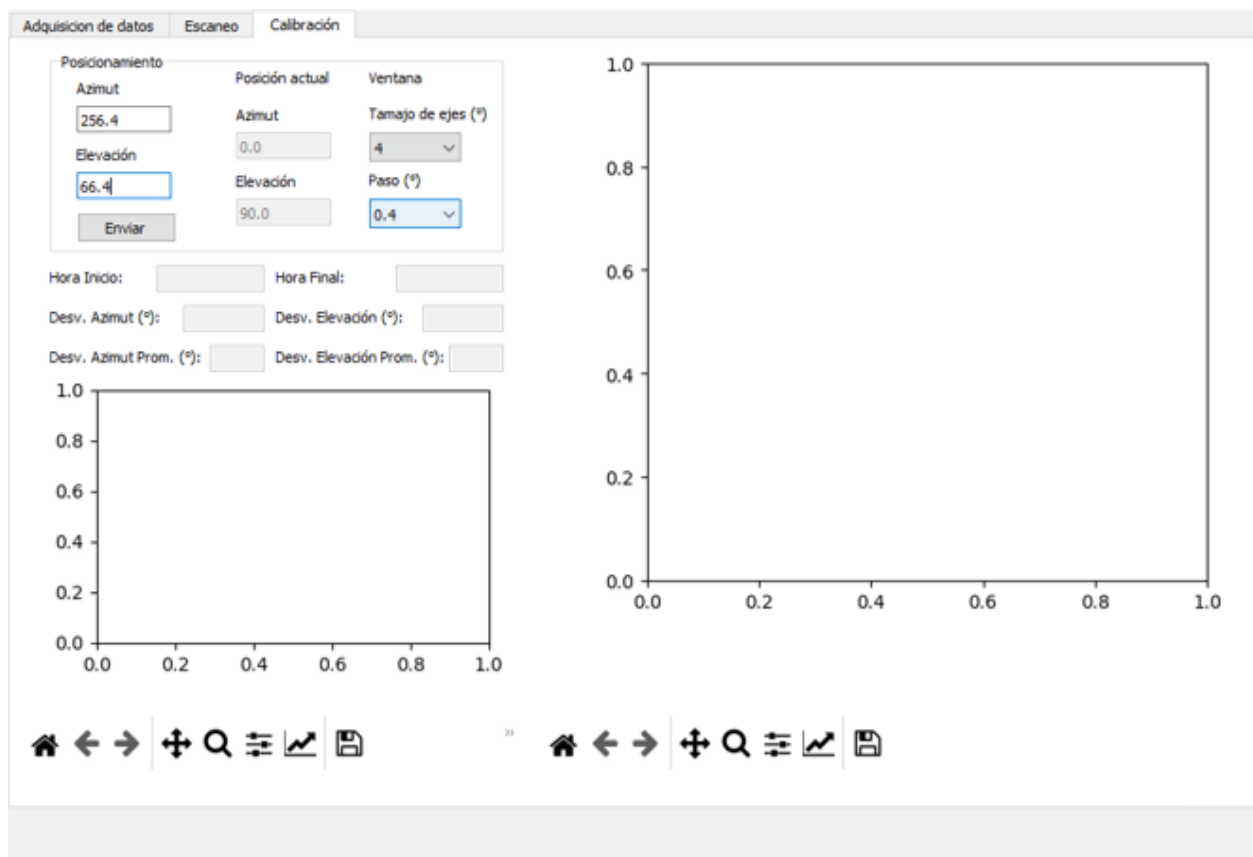


Figura B.14: Ingreso de parámetros

El resultado final se observa en la Figura B.15. Al culminar proceso de adquisición, en la parte derecha de la ventana, se visualiza la representación gráfica de la zona observada, donde cada pixel de la imagen representa la posición donde fue tomado el dato. Además, en la parte izquierda se puntualizan la hora de inicio y finalización proceso. La desviación entre el punto de mayor detección y la posición central de la matriz se obtiene en cada sesión, permitiendo visualizar la magnitud de este desfase en cada cada dirección del movimiento en la zona izquierda de la ventana.

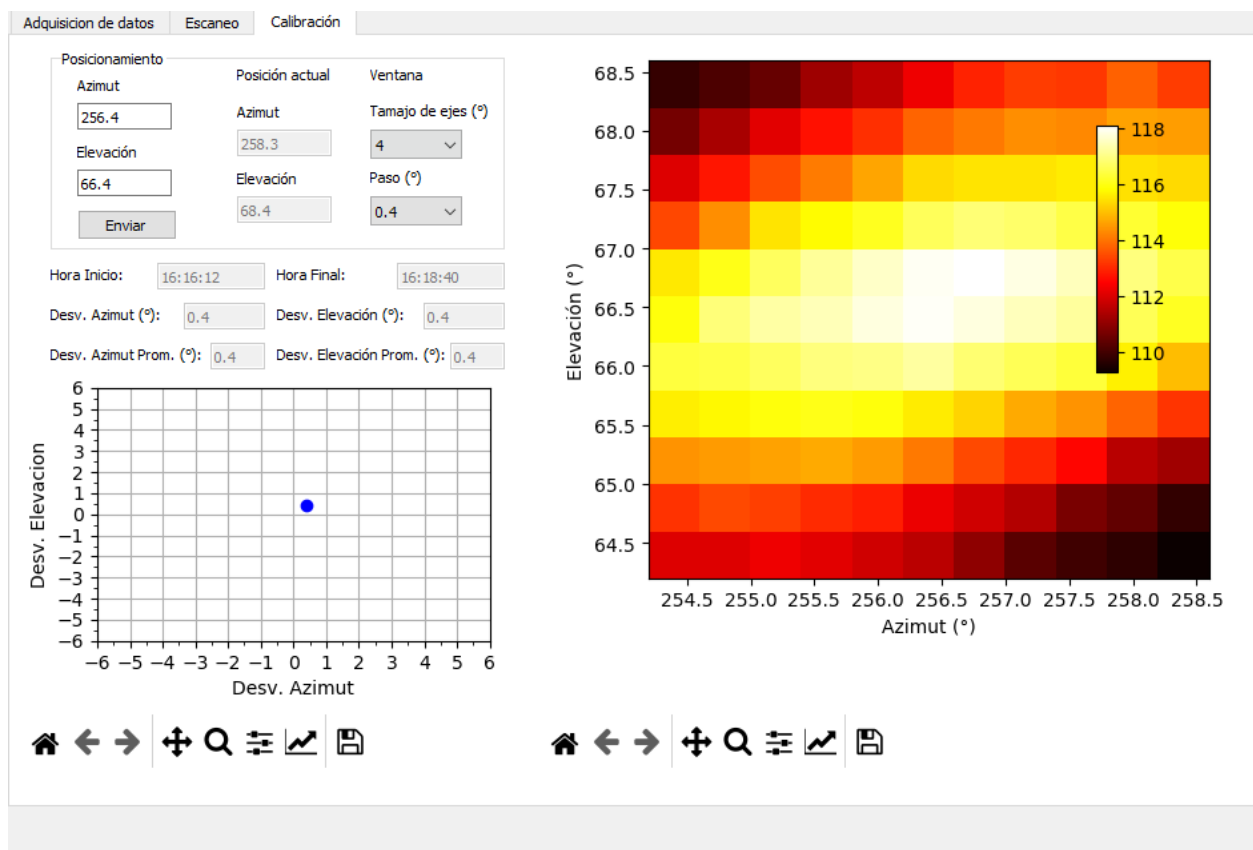
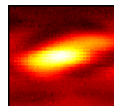


Figura B.15: Resultado final pestaña Calibración

Para realizar un correcto proceso de calibración es necesario realizar múltiples sesiones de adquisición alrededor del mismo objeto. Esto permitirá realizar una mejor aproximación al error de apuntado al calcular la desviación promedio en ambas direcciones de movimiento. Como se observa en la Figura B.16, el promedio de desviación se presenta como valor numérico en la zona izquierda de la ventana principal, mientras que en la zona inferior se grafica el valor de la desviación durante cada sesión.

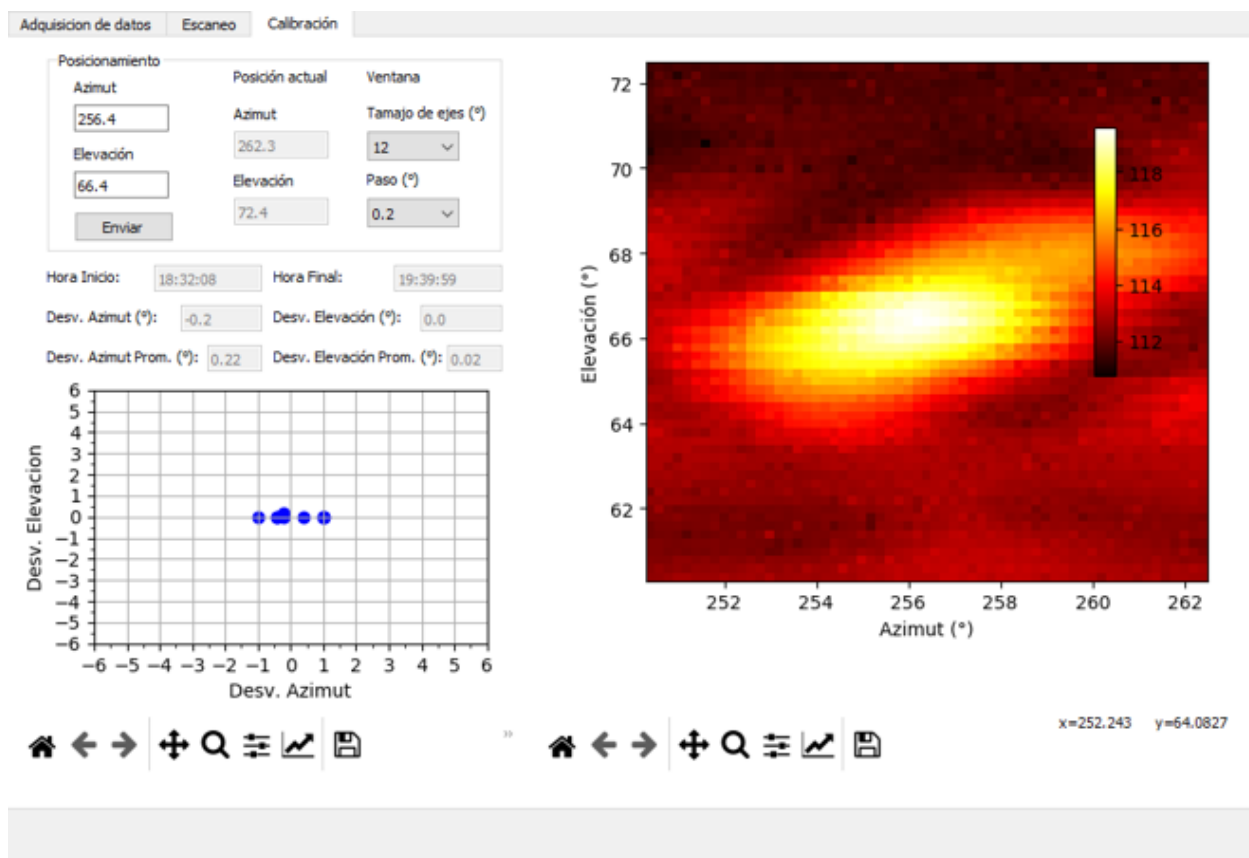
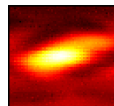


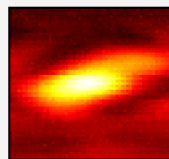
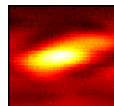
Figura B.16: Resultado final pestaña Calibración tras varias sesiones de adquisición.

3.4. Menú de Ayuda

En este menú se encuentran alojados los botones “Acerca De” y “Guía de Usuario”.

3.4.1. Acerca De

Al ejecutar esta acción, la interfaz gráfica de Asteroid Position despliega una ventana en la que se presentan la versión del software, los autores, la filiación institucional de los mismos, y la fecha de finalización del desarrollo (ver Figura B.17).



Compact Radio Telescope

Versión 1.0

David Galeano Gómez
Edwin Andrés Quintero Salazar

Grupo de Investigación en Astronomía Alfa Orión

Universidad Tecnológica de Pereira

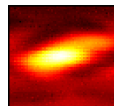
Noviembre de 2020



Figura B.17: Ventana “Acerca de”.

3.4.2. Guía de Usuario

Al accionar este botón el software despliega este documento de ayuda.



4. Ejemplos de aplicación

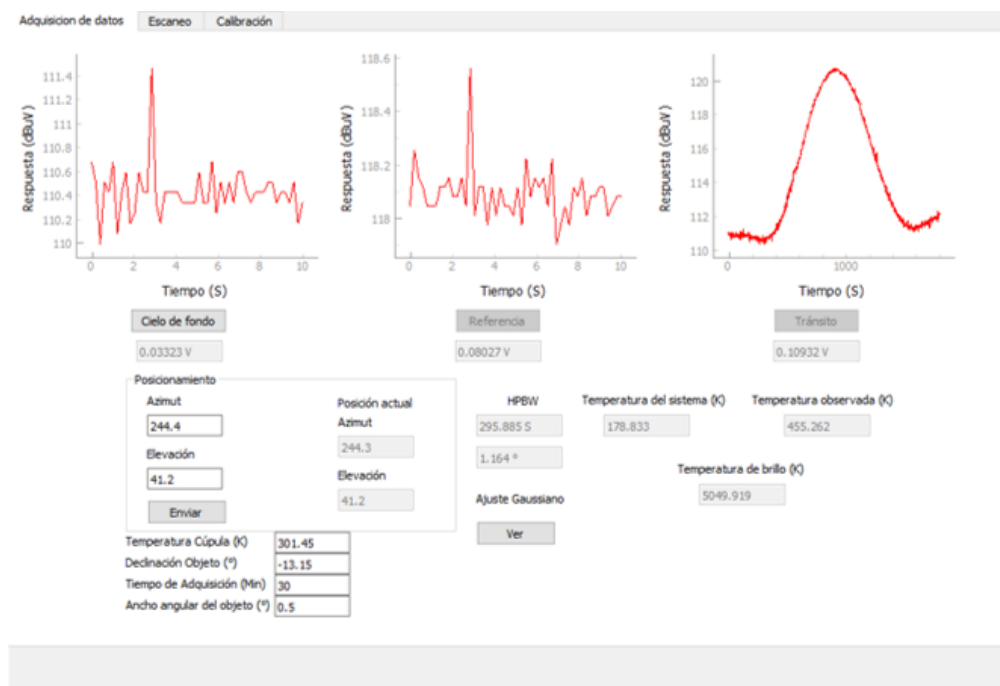


Figura B.18: Ventana Adquisición de datos al apuntar el sol el 3/11/2020 a las 14:33.

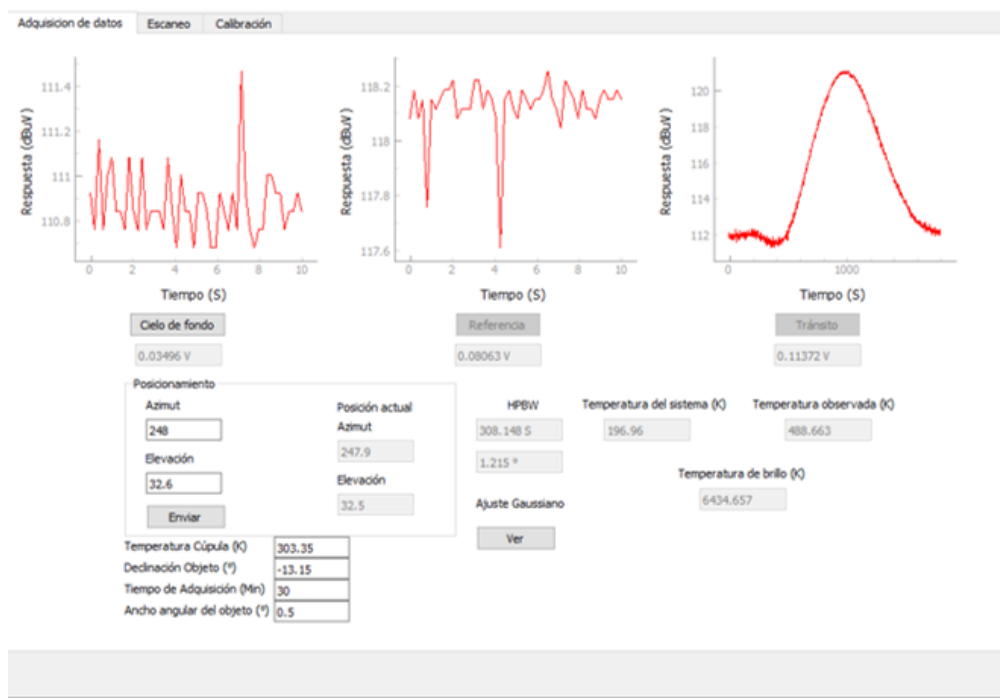
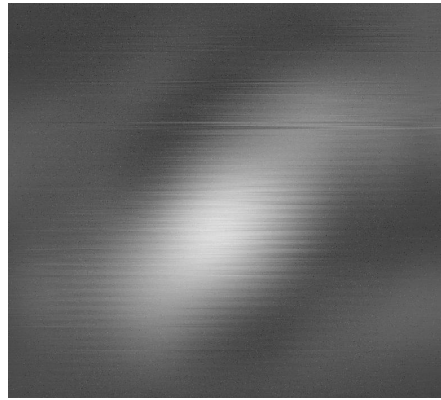
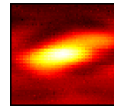
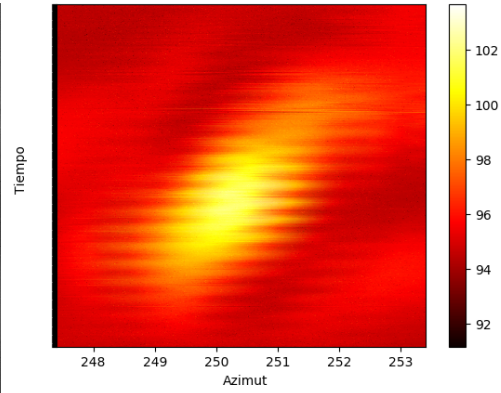


Figura B.19: Ventana Adquisición de datos al apuntar el sol el 3/11/2020 a las 15:07.

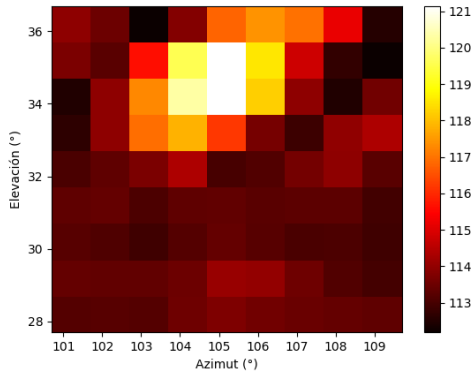


(a) Blanco y negro.

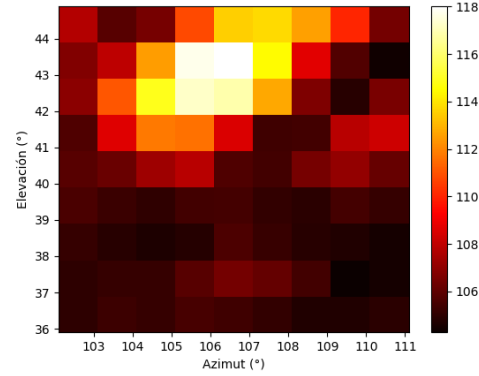


(b) Mapeo de color.

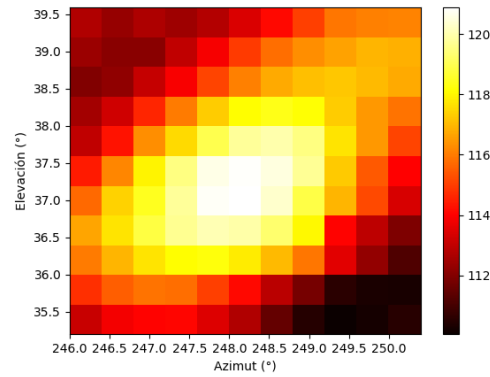
Figura B.20: Detección del sol con la ventana Escaneo.



(a) $12^\circ \times 12^\circ$ y pasos de 1° .

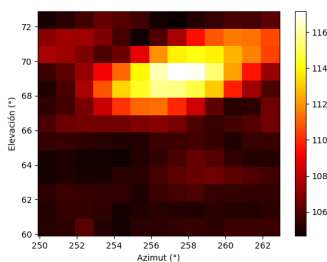
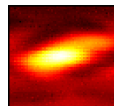


(b) $12^\circ \times 12^\circ$ y pasos de 1° .

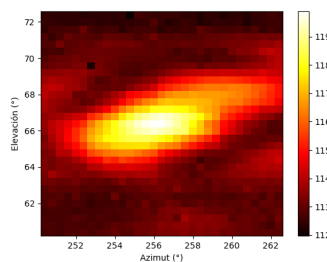


(c) $4^\circ \times 4^\circ$ y pasos de 0.4° .

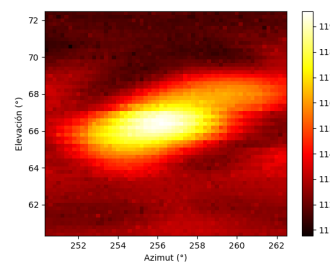
Figura B.21: Ventanas de calibración alrededor del Sol.



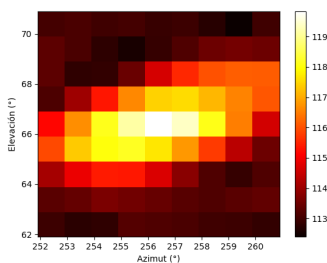
(a) $12^\circ \times 12^\circ$ y pasos de 1°



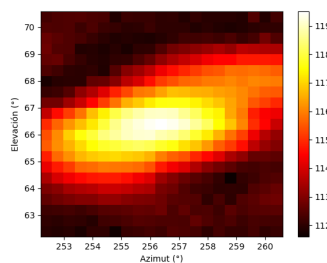
(b) $12^\circ \times 12^\circ$ y pasos de 0.4°



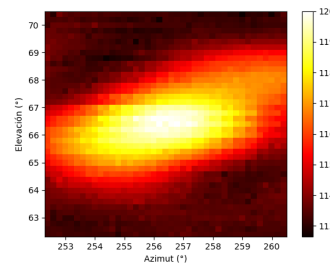
(c) $12^\circ \times 12^\circ$ y pasos de 0.2°



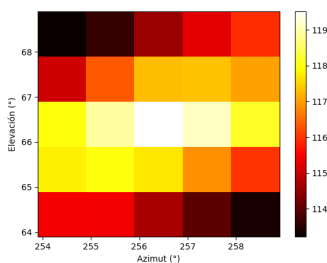
(d) $8^\circ \times 8^\circ$ y pasos de 1°



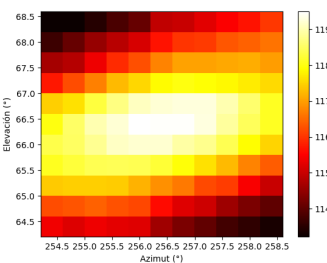
(e) $8^\circ \times 8^\circ$ y pasos de 0.4°



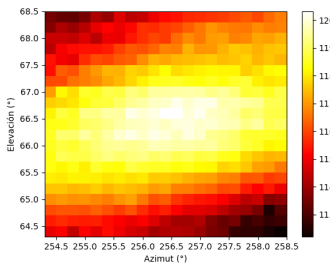
(f) $8^\circ \times 8^\circ$ y pasos de 0.2°



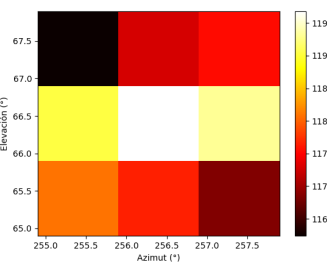
(g) $4^\circ \times 4^\circ$ y pasos de 1°



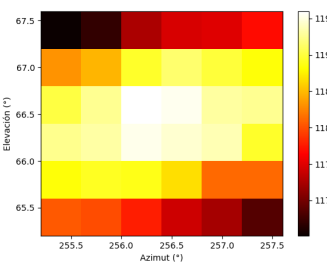
(h) $4^\circ \times 4^\circ$ y pasos de 0.4°



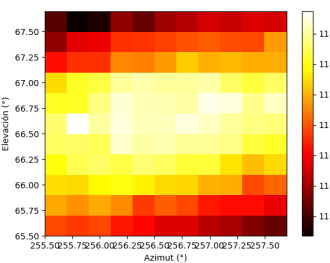
(i) $4^\circ \times 4^\circ$ y pasos de 0.2°



(j) $2^\circ \times 2^\circ$ y pasos de 1°



(k) $2^\circ \times 2^\circ$ y pasos de 0.4°



(l) $2^\circ \times 2^\circ$ y pasos de 0.2°

Figura B.22: Ventanas de calibración alrededor de Intelsat 30.

5. Algoritmos

5.1. Ventana Adquisición de datos

El algoritmo que define el funcionamiento de la página se enfoca entonces en optimizar las sesiones de adquisición, definiendo un orden para la toma de datos. Una vez inicializada la ventana, se habilita el ingreso de la coordenada a la que deseamos enviar el radiotelescopio en la caja “Posicionamiento”. En esta se habilitan las cajas de texto para el ingreso del acimut y altitud donde queremos apuntar la antena, durante cualquier momento de la adquisición. Una vez es oprimido el botón “Enviar”, se ejecuta la función *SendPosition()* para las coordenadas ingresadas. Al completar el movimiento, se actualiza la posición como la coordenada final de la antena utilizando la función *Position()*. Por otro lado, al iniciar encontraremos desbloqueado el botón “Cielo de Fondo”, el cual cumple la función de captar la respuesta del radioreceptor durante cinco segundos utilizando la función *ADQ()*. Por último, se construye la gráfica bidimensional de los datos tomados y se muestra el valor en voltios como se ilustra en el diagrama de flujo de la Figura B.23.

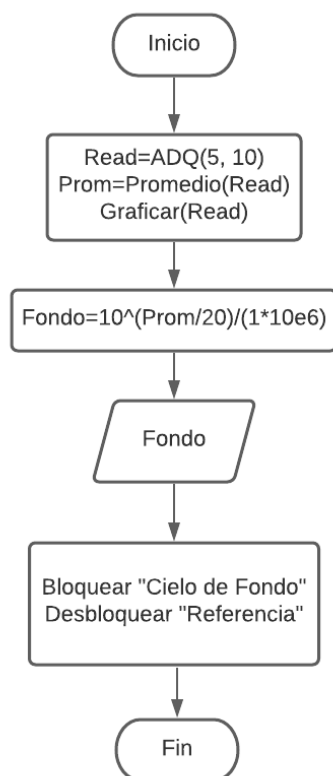
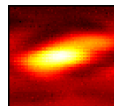


Figura B.23: Diagrama de flujo al oprimir el botón “Cielo de fondo”.



Una vez finalizado el proceso se desbloquea el botón “Referencia”. Este cumple la misma función del anterior, pero en este caso se debe realizar la adquisición de datos al apuntar un objeto cuya temperatura sea conocida (ver Figura B.24).

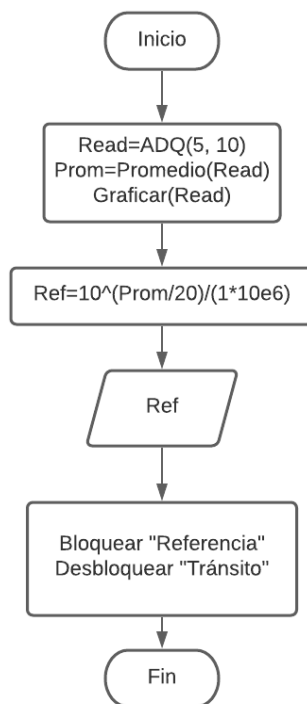
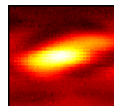


Figura B.24: Diagrama de flujo al oprimir el botón “Referencia”.

Finalmente se habilitará el botón “Tránsito”. Antes de comenzar la captura del tránsito es necesario ingresar los siguientes parámetros.

- **Dec:** Declinación del objeto observado.
- **Time:** Tiempo de adquisición.
- **Tc:** Temperatura del objeto de referencia.
- **Ancho:** Ancho angular del objeto observado.

Al oprimir el botón, se ejecuta el algoritmo de la Figura B.25, realizando adquisición de datos durante el tiempo ingresado. Una vez finalizada la sesión de adquisición, se guarda la información obtenida en un archivo excel. En este punto se utiliza la información adquirida en los tres momentos, con la cual se calcula la temperatura del sistema (T_{sys}) y la temperatura observada por la antena (T_A). Para complementar estos resultados se extrae el ancho de haz a partir del patrón de radiación (HBPWC) obtenido en el tránsito, el cual deriva en



el área iluminada y eficiencia del plato reflector. Finalmente, se utilizan estos valores para determinar la temperatura de brillo del objeto y así terminar la sesión de adquisición.

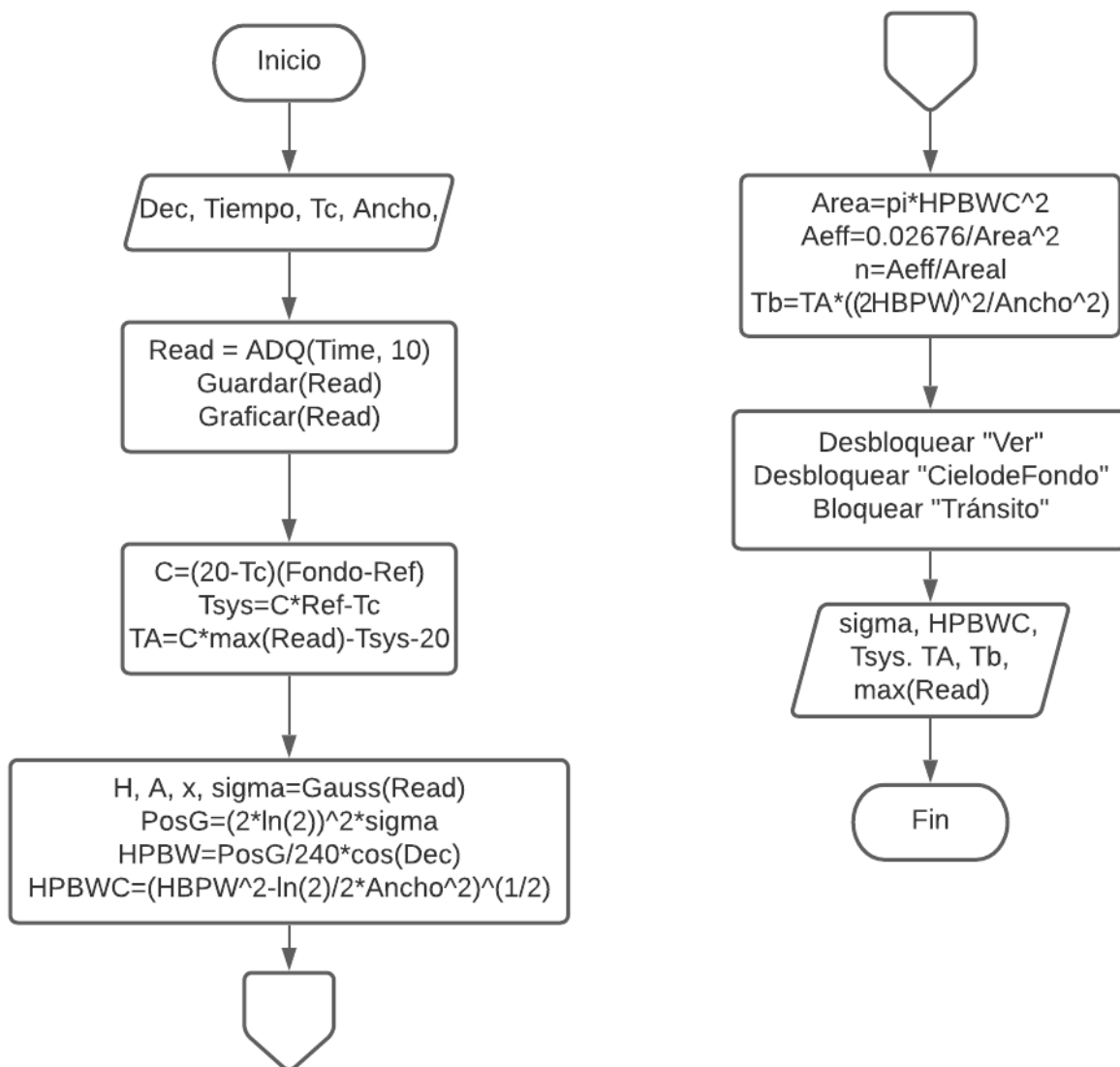
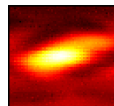


Figura B.25: Diagrama de flujo al oprimir el botón “Tránsito”.



Para verificar la fidelidad de la aproximación Gaussiana con los datos obtenidos, se habilita el botón “Ver”, el cual permite visualizar el ajuste Gaussiano implementado en el cálculo del ancho de haz de la antena. Al oprimir el botón se ejecuta el algoritmo mostrado en el diagrama de flujo de la Figura B.26.

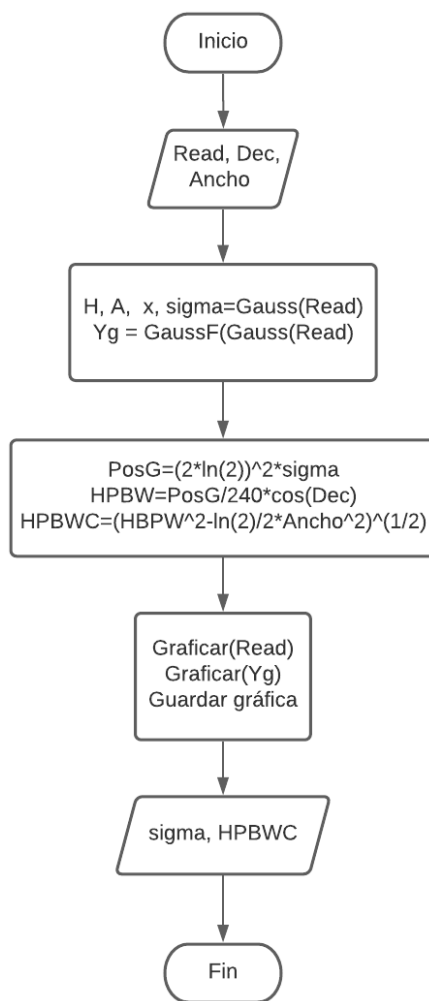


Figura B.26: Diagrama de flujo al oprimir el botón “Ver”.

5.2. Ventana Escaneo

Ya que el algoritmo de la ventana se encarga de generar una imagen de una zona del cielo al presionar el botón “Enviar”, las coordenadas ingresadas serán el punto central de la misma. Para realizar una sesión de adquisición óptima es necesario ingresar los parámetros que definen el ancho y alto de la imagen, además de la coordenada central a observar. Antes de iniciar se deben ingresar los siguientes valores.

- **Picsize:** Altura de la imagen en pixeles.
- **Sample:** Frecuencia de muestreo durante cada movimiento.
- **Ancho:** Ancho angular de la ventana.
- **Time:** Tiempo tomado por la antena para realizar el movimiento.

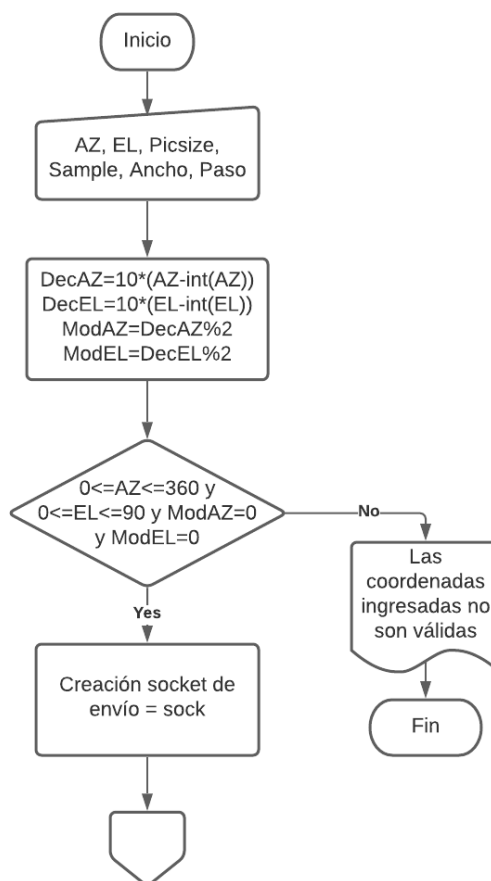


Figura B.27: Ventana Escaneo: Primera parte.

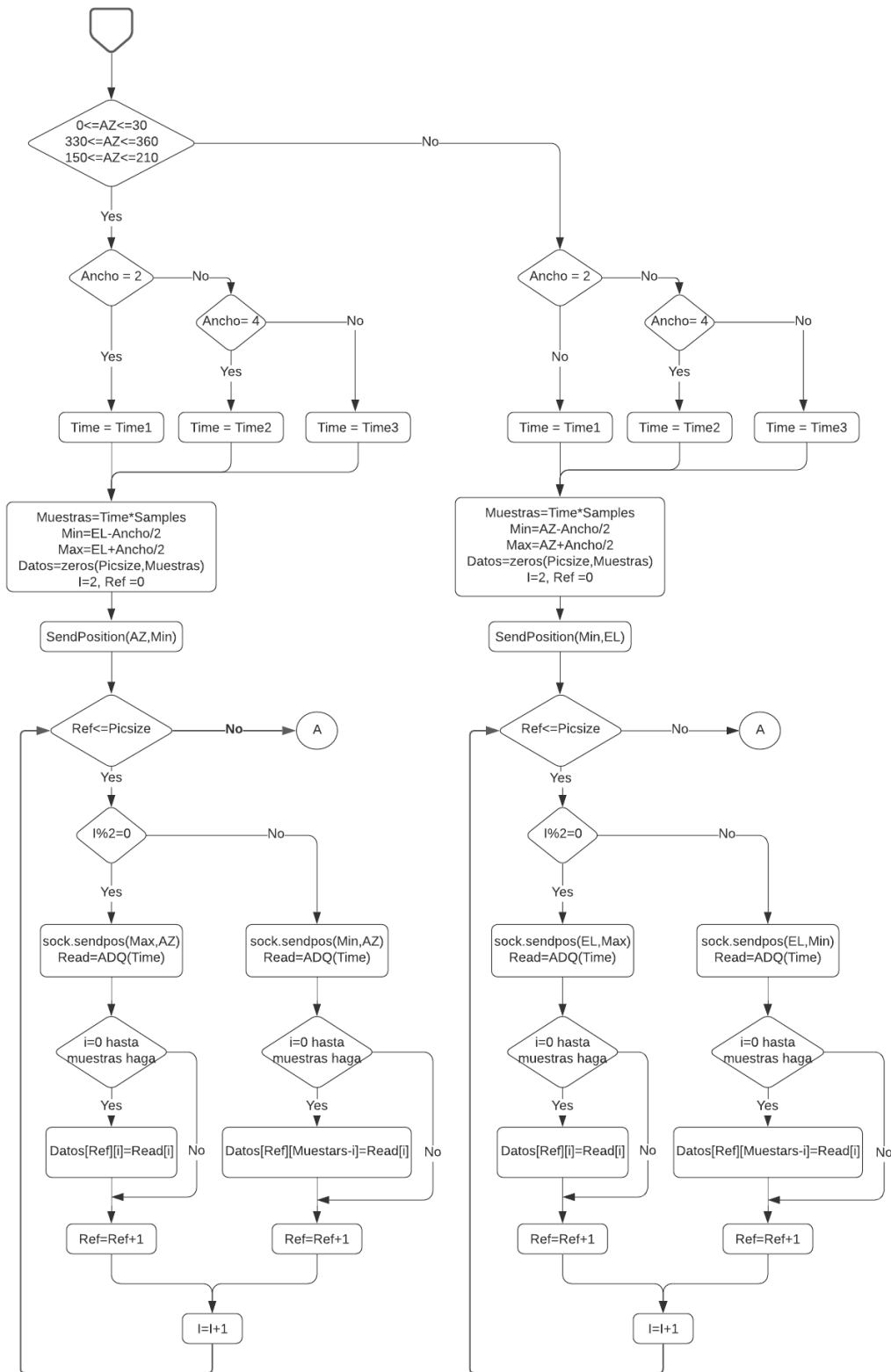
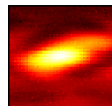
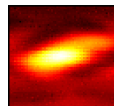


Figura B.28: Ventana Escaneo: Segunda parte.



Dado que este proceso implica dejar una de las coordenadas constante mientras la otra realiza un movimiento cíclico de barrido, esperando captar el objeto cuando este pase por el haz de la antena, la elección de la coordenada que se mantendrá constante dependerá de la coordenada de acimut en la que se ubicará el radiotelescopio. De esta forma, aquellas coordenadas con un acimut en los rangos de $0^\circ - 30^\circ$, $150^\circ - 210^\circ$, $330^\circ - 360^\circ$, realizarán la adquisición manteniendo el acimut constante, mientras que, aquellas que no cumplan esta condición, realizarán la adquisición manteniendo la altitud constante. El proceso se puede realizar con un ancho de ejes de 2° , 4° y 6° durante un tiempo de adquisición sincrónico con el movimiento. El envío de las posiciones deseadas se logra a través de la función *SendPosition()*. Teniendo en cuenta que, el tiempo de adquisición debe ser sincrónico con el tiempo del movimiento, es necesario ingresar el tiempo medido para el movimiento de la antena en cada dirección y magnitud. Esta sección del algoritmo general se muestra en las Figuras B.27 y B.28. Para la lectura de datos se implementa la función *ADQ()*, teniendo como variable de entrada el tiempo escogido por el usuario para el movimiento.

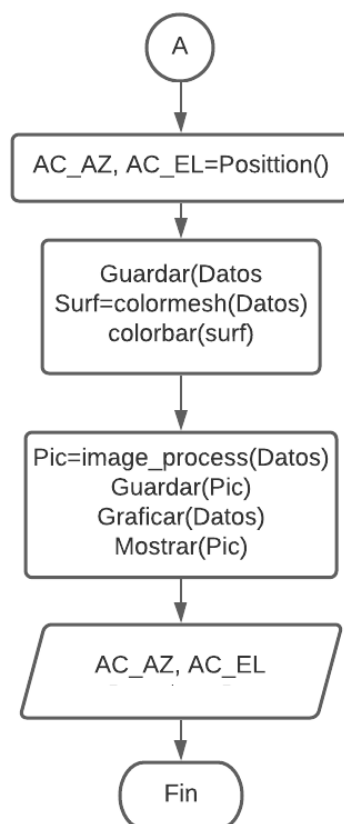
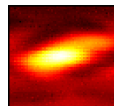


Figura B.29: Ventana Escaneo: Tercera parte.



Una vez finalizada la adquisición, se definen, en el algoritmo de la Figura B.29, los límites de los ejes bidimensionales de la gráfica, a partir de las posiciones de la antena y el tiempo de cada movimiento, para luego ser mostrada con un mapeo de color. Para la construcción del mapa bidimensional, que permite contrastar las intensidades captadas, se hace uso del módulo *Pcolormesh()*, extraído de la librería *Matplotlib*¹. Se establece entonces el eje horizontal como la coordenada en variación y el eje vertical a partir del momento en que se realizó el movimiento. Además, utilizando el módulo *Colorbar()*, se construye una barra de colores en representación al valor en intensidad de cada una de las posiciones para la figura gráfica. Por otro lado, para realizar la construcción de una imagen a partir de datos almacenados por las sesiones de adquisición, se utilizó el módulo *Image*, de la librería *Pillow*². Este permite generar una imagen en escala de grises a partir de los datos normalizados, además facilita su almacenamiento y lectura.

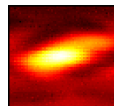
5.3. Ventana Calibración

Esta ventana de la aplicación permite realizar la observación del cielo a través de una matriz cuadrada de tamaño predefinible al presionar el botón “Enviar”. En esta, cada punto representa un valor de intensidad medido para la posición en la que se encuentra apuntado el radiotelescopio. Como variable de ingreso se tomará el ancho de la ventana (Ancho), la resolución que esta tendrá (Paso) y la posición donde se enviará el radiotelescopio. Una vez se han ingresado el tamaño de la ventana y los pasos con que se ejecutará esta, se definen los puntos coordinados mínimo y máximo, enviando la antena al punto central donde se adquiere el primer valor número de la matriz como se muestra en la Figura B.30.

Una vez se ha definido el tamaño de la matriz se pasa a generar una posición para cada elemento de esta. Así, cada vez que el telescopio cambia de posición, es guardado en el elemento consecuente de la matriz el valor de la intensidad medido. Por otro lado, el recorrido espacial a realizar dependerá de la posición ingresada. De esta forma se realiza la adquisición en una dirección de altitud descendente cuando el acimut ingresado es menor a 180 °, mientras que las coordenadas mayores a este valor son correspondientes a una adquisición en altitud ascendente. En ambos casos se toma la coordenada central como punto de inicio. Como se muestra en las Figura B.31, cuando la condición se cumple, el radiotelescopio se envía a la posición máxima de acimut y elevación, iniciando el recorrido de la matriz definida por el usuario, saltando la adquisición en la coordenada central, hasta llegar a la posición mínima de altitud de la matriz ingresada. Por otro lado, cuando la condición no es cumplida, se

¹<https://matplotlib.org/>

²<https://pillow.readthedocs.io/en/stable/>



realiza el protocolo ilustrado en la Figura B.32, iniciando el recorrido en la posición mínima de acimut y altitud, saltando la adquisición en la coordenada central y culminando en el punto máximo de altitud.

Una vez finalizada la sesión de adquisición se guardan los datos y se definen los límites de los ejes bidimensionales de la gráfica resultante, a partir de las posiciones máxima y mínima de la antena. Se construye la malla bidimensional para ser mostrada en un mapeo de color utilizando nuevamente las funciones *Pcolormesh()* y *Colrobar()*. Finalmente, se extrae la desviación entre el punto de mayor detección y la posición central de la ventana ejecutada, almacenando este valor constantemente para llevar así su valor promedio. De forma complementaria, se ilustra la desviación en cada ventana de observación efectuada en una gráfica bidimensional utilizando la función *Scatter()* de la librería *Matplotlib*. El algoritmo que define el protocolo final se ilustra en la Figura B.33.

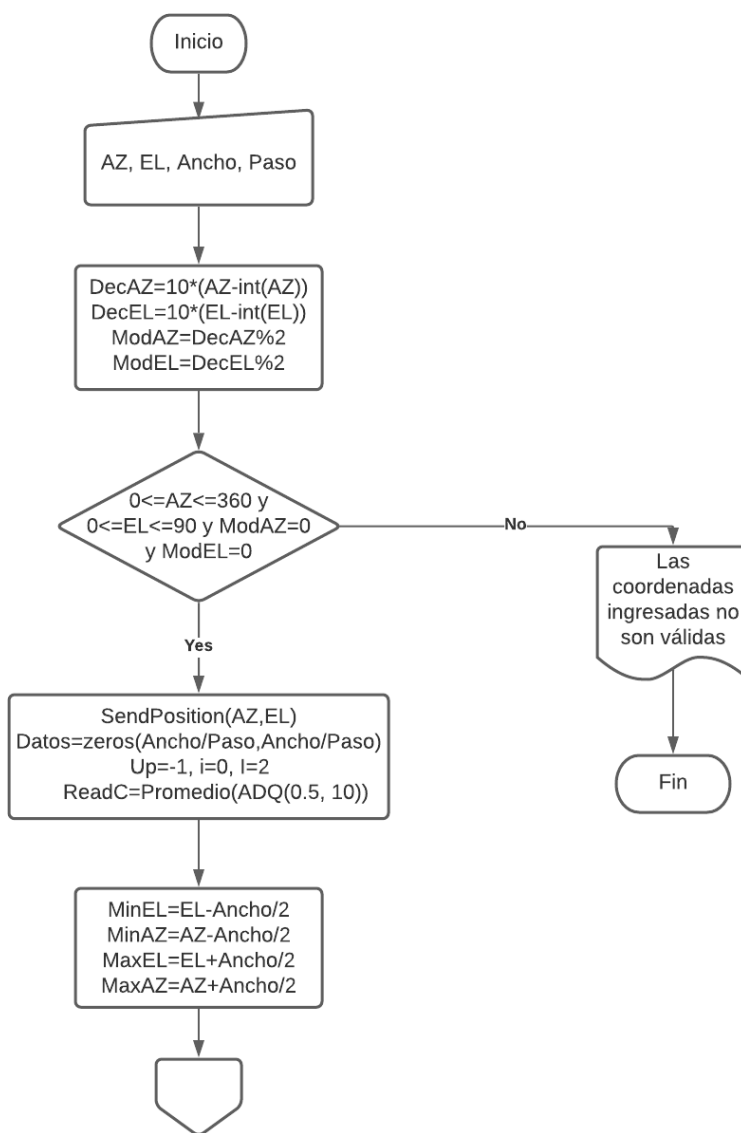
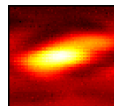


Figura B.30: Ventana Calibración: Primera parte.

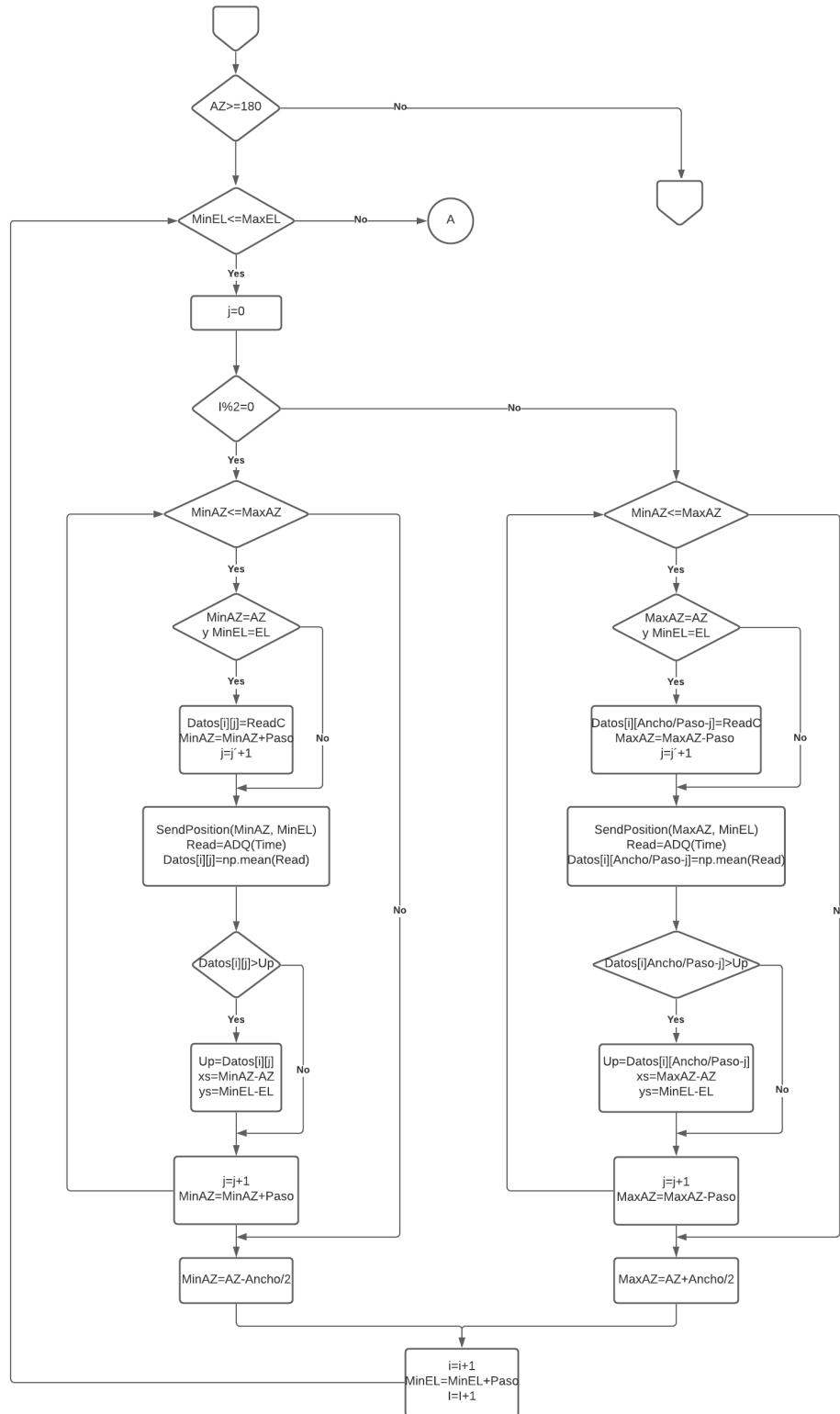
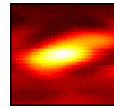


Figura B.31: Ventana Calibración: Segunda parte.

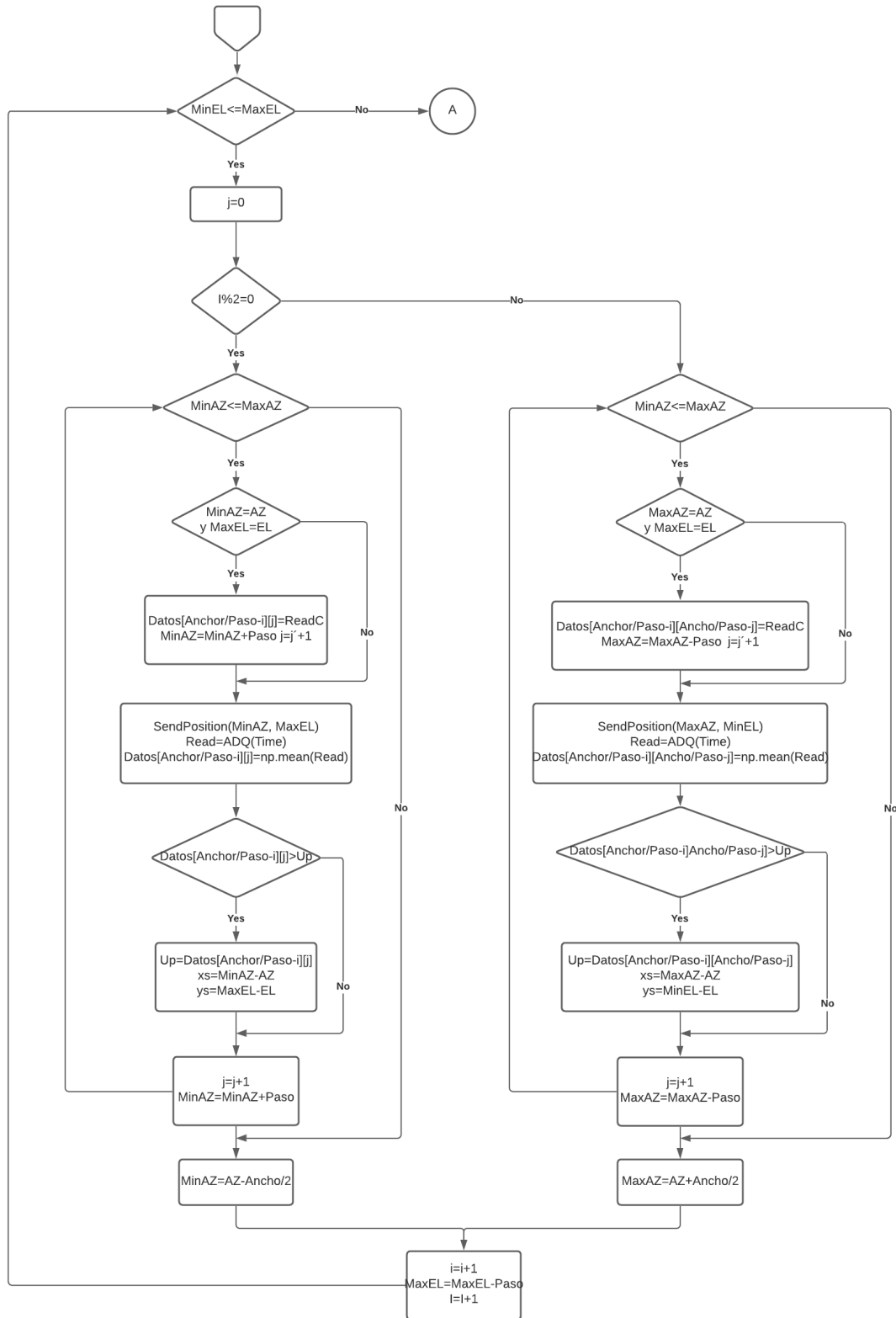
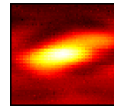


Figura B.32: Ventana Calibración: Tercera parte.

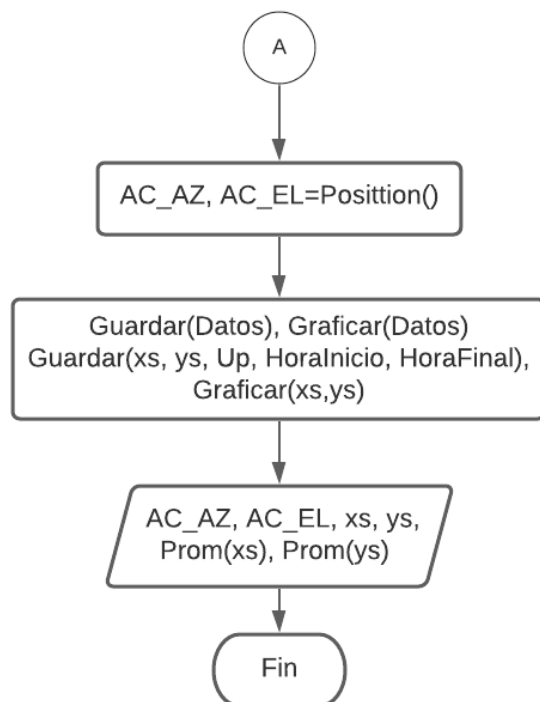
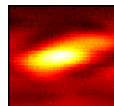


Figura B.33: Ventana Calibración: Cuarta parte.

5.4. Comunicación DAQ

Como parte inicial de la construcción de la interfaz se realizó la función encargada de captar y procesar el voltaje recibido por medio de la tarjeta de adquisición de datos. Para esto se implementó la librería *NIDAQmx*³, la cual permite configurar el tiempo de adquisición y la frecuencia de muestreo fácilmente. Utilizando la función *Task.read()*, se logró la adquisición de datos durante los parámetros configurados. Una vez almacenada la información, esta es escalada, pasando su magnitud de V a $dBuV$, para luego ser procesada a través de un filtro Butterworth pasa bajos digital de segundo orden, con una frecuencia de corte de 0.5 Hz. Este proceso se logra al implementar las funciones *Butter()* y *filtfilt()* de la librería *Scipy*⁴. Cabe resaltar que el filtrado solo se realiza para señales con una duración mayor a 5 s. El algoritmo completo se almacenará en la función *ADQ()*, cuyo diagrama de flujo es ilustrado en la Figura B.34. Esta tendrá como variable de entrada el tiempo de adquisición, dando en su salida el arreglo completo de los datos procesados.

³<https://nidaqmx-python.readthedocs.io/en/latest/>

⁴<https://www.scipy.org/>

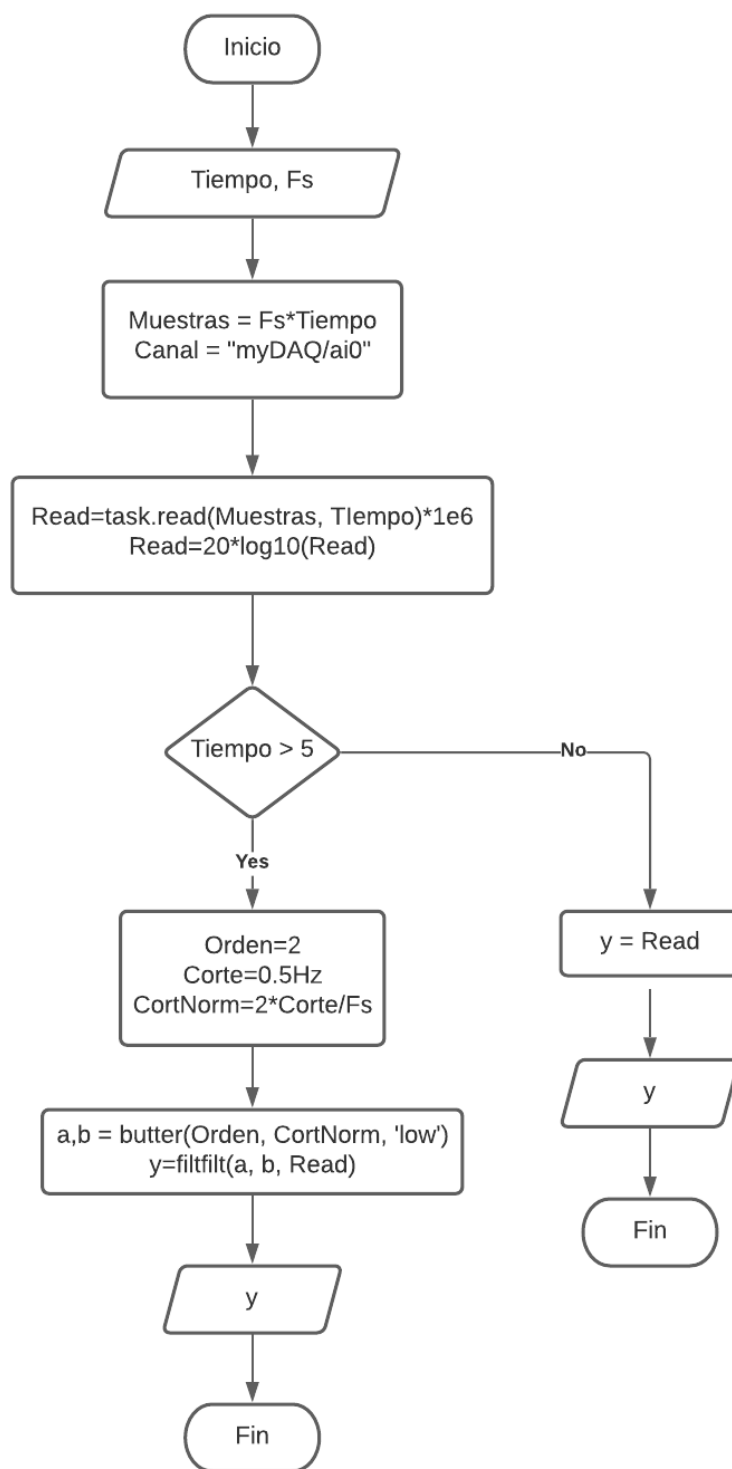
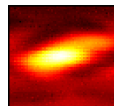


Figura B.34: Diagrama de flujo de la función *ADQ()*.

5.5. Envío y recepción de coordenadas

Para el envío y recepción de coordenadas es necesaria la adquisición del programa Pstrotator (Figura B.35), diseñado para múltiples monturas con movimiento en acimut y altitud. La principal ventaja de este programa es la capacidad de comunicarse a través de protocolos de datagramas de usuario (UDP). Una vez activada la comunicación UDP es posible enviar y recibir coordenadas a la montura a través de programas externos.

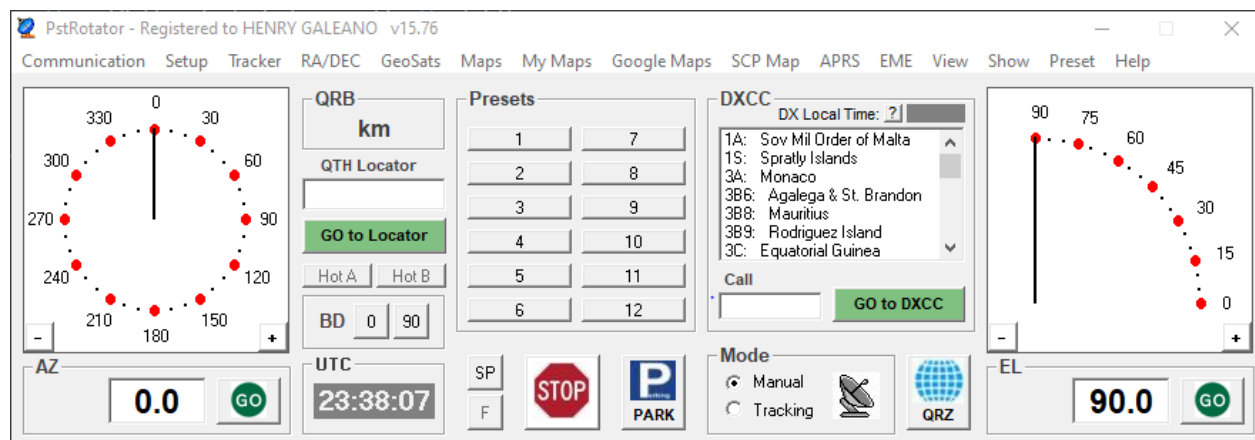


Figura B.35: Ventana principal Pstrotator.

Teniendo habilitada la comunicación UDP con Pstrotator, se construyeron las funciones en Python para el envío y recepción de datagramas. Como primera instancia se realizó la función encargada de recibir y reportar la posición de la montura como se muestra en el diagrama de flujo de la Figura B.36. En ella se utiliza la librería *Socket*⁵, agilizando la comunicación a través del sistema UDP. Se inicia entonces con la creación de los sockets para el envío y recepción de coordenadas con los puertos UDP 12000 y 12001 respectivamente. El primer socket se encarga de enviar el comando para solicitar el reporte de la posición actual al canal 12000, mientras que, el segundo escucha la respuesta en el canal 12001. Esta combinación permite realizar la adquisición de las coordenadas del radiotelescopio siempre que se utiliza la función *Position()*.

⁵<https://docs.python.org/3/library/socket.html>

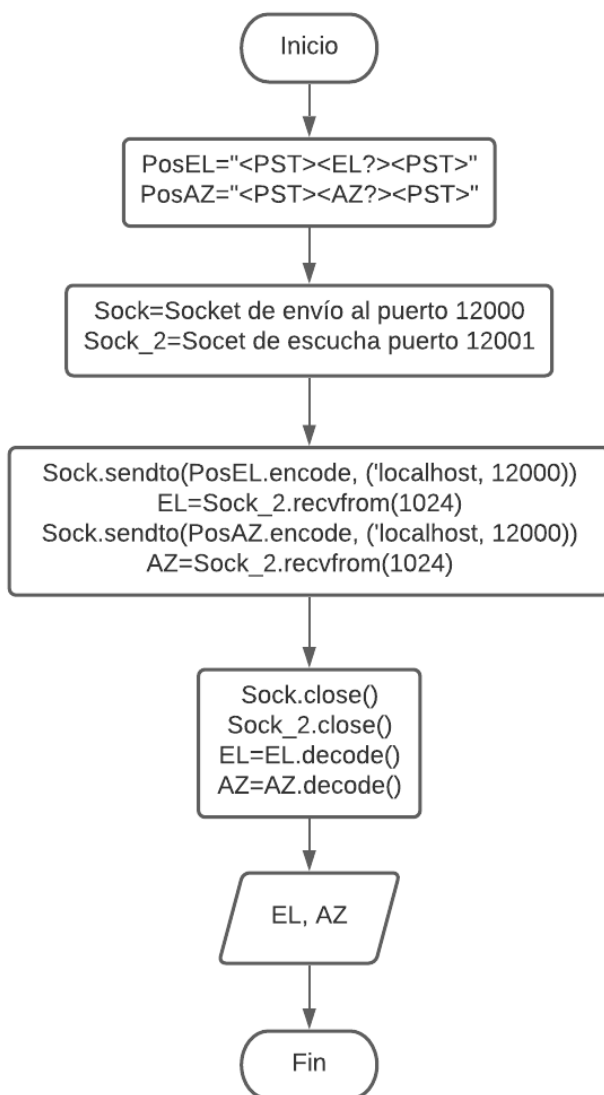
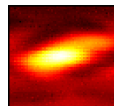


Figura B.36: Diagrama de flujo función *Position()*.

Ahora, para el envío de coordenadas se construyó la función *SendPosition()*, mostrada en las Figuras B.37 y B.38. Esta toma los valores de acimut y altitud ingresadas por el usuario, validando que se encuentren bajo el rango permitido y así realizar el proceso de envío. Nuevamente se crean los sockets de envío y recepción, conectados a los puertos 12000 y 12001. El primer socket se encargará de enviar la coordenada deseada a PstRotator, para luego escuchar el puerto de actualización constantemente con el segundo socket, deteniendo el proceso cuando la montura haya alcanzado la posición deseada. Una vez culminado el algoritmo de la función, esta da como resultado la posición final del radiotelescopio.

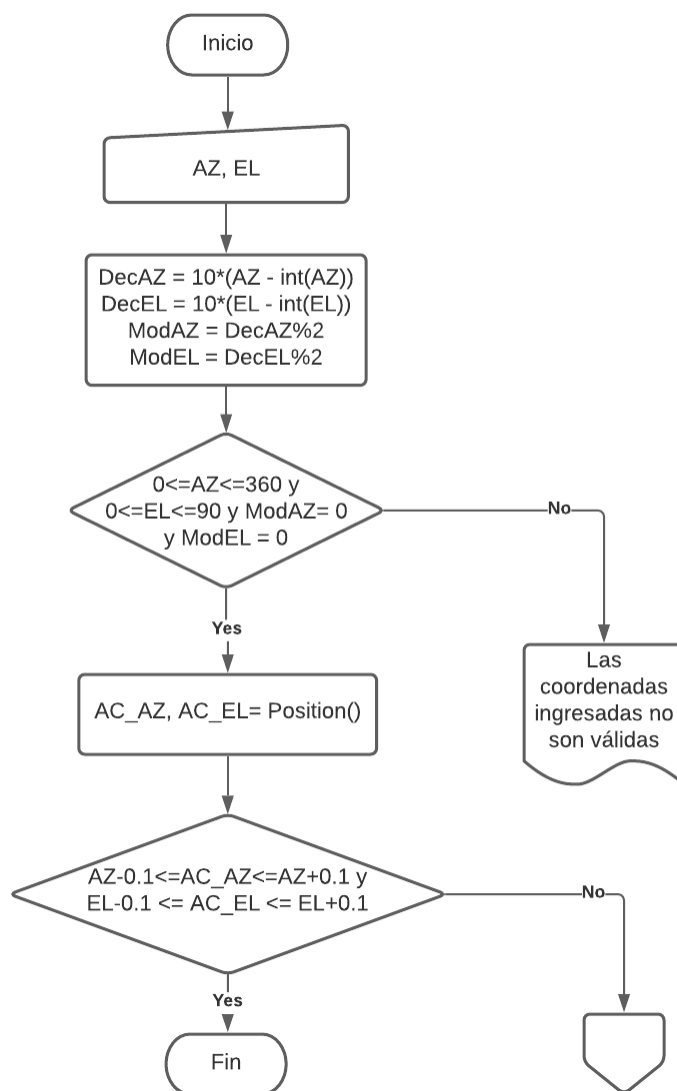
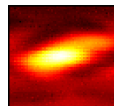


Figura B.37: Función *SendPosition()*: Primera parte.

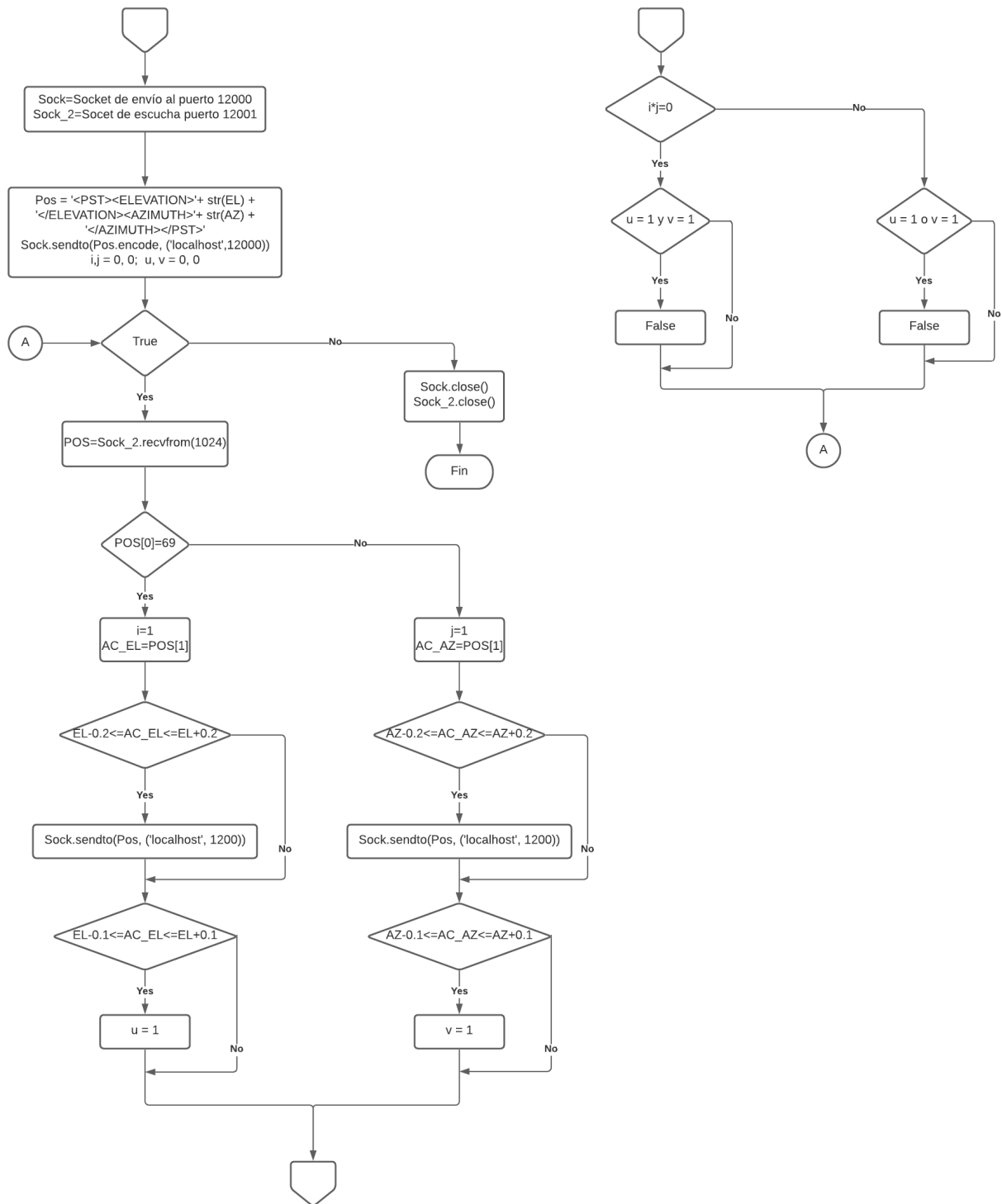
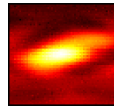


Figura B.38: Función *SendPosition()*: Segunda parte.

5.6. Ajuste Gaussiano

Teniendo en cuenta que el cálculo del ancho medio de haz de la antenna implica la interpretación de la curva gaussiana resultante de un tránsito solar, se construyó la función *Gauss()*. Como se observa en el diagrama de flujo de la Figura B.39, esta entrega como resultado los parámetros de la función gaussiana más cercana a los datos ingresados. En la construcción del ajuste Gaussiano de un arreglo de datos, se implementó la función *Optimize()* de la librería Scipy. Antes de ser utilizada es necesario hallar el valor promedio de los valores a optimizar, además de su varianza. Posteriormente se ingresan estos valores a la función *Optimize()*, junto con la función esperada, en nuestro caso una campana de Gauss definida como $GaussF(x, H, A, x0, sigma) = H + Ae^{\frac{(x-x0)^2}{sigma^2}}$. Esta función se llamará *GaussF()*. Finalmente, se obtiene como resultado la interpolación más cercana a los datos ingresados con la función *optimize.curve_fit()*, la cual entrega las constantes *H*, *A*, *x0*, *sigma* optimizadas para el arreglo ingresado.

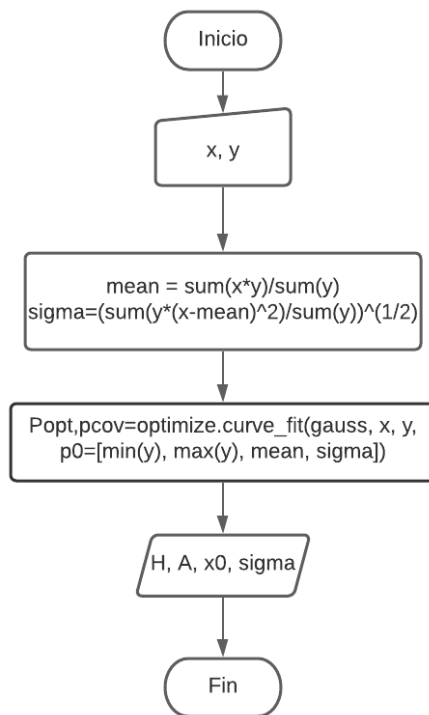
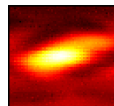


Figura B.39: Diagrama de flujo *Gauss()*.



Bibliografía

Chanka, N. (2018). Development of an off the shelf methanol maser telescope. Master's thesis, School of Physics , Astronomy and Mathematics.

de Parter, I. y S. Kurth, W. (2014). Chapter 52 - the solar system at radio wavelengths. In Spohn, T., Breuer, D., y Johnson, T. V., editors, *Encyclopedia of the Solar System (Third Edition)*, pages 1107 - 1132. Elsevier, Boston, third edition.

Gaylard, M. (2012). *Radio Astronomy with a Satellite Dish*. South African Radio Astronomy Observatory.

Gulkis, S. y de Pater, I. (2003). Radio astronomy, planetary. In Meyers, R. A., editor, *Encyclopedia of Physical Science and Technology (Third Edition)*, pages 687 712. Academic Press, New York, third edition edition.

Holleman, I. y Beekhuis, H. (2004). Weather radar monitoring using the sun. Technical report, Royal Dutch Meteorological Institute.

Mandal, P., Agarwal, D., Kumar, P., Yelikar, A., Soni, K., y Krishna, V. (2016). Everyday radio telescope.

O'Neil, K. (2001). Single dish calibration techniques at radio wavelengths. *In ASP Conference Series*.

B.2 Código fuente funciones estructurales “CRT.py”

```
import sys, os
#Librería comunicación UDP:
import socket

import time
import math
#Librería comunicación NiDAQ:
import nidaqmx

import PIL
#Librerías para el almacenamiento de datos en excel:
import pandas as pd
import xlsxwriter

#Librerías para el la optimización Gaussiana
from scipy import optimize
from scipy.signal import butter, lfilter, freqz, filtfilt

from PIL import Image, ImageQt

from nidaqmx.constants import Edge, AcquisitionType
from nidaqmx import stream_readers, constants

import numpy as np

from PyQt5 import QtWidgets, uic, QtGui, QtCore
from PyQt5.uic import loadUi
from PyQt5.QtWidgets import *
from PyQt5.QtGui import *
from PyQt5.QtCore import *
from PyQt5.QtWinExtras import QtWin
#Adjunta graficador pyqtgraph
import pyqtgraph as pg
from pyqtgraph import PlotWidget, plot
```



```

#Lee el entorno gráfico de Adquisicion.py
from Adquisicion import Ui_MainWindow
import resources

#Adjuntar graficador Matplotlib
import matplotlib
import matplotlib.pyplot as plt
matplotlib.use('TkAgg')
from matplotlib.dates import DateFormatter, HourLocator
from mpl_toolkits.mplot3d import Axes3D # noqa: F401 unused import
from matplotlib.ticker import LinearLocator, FormatStrFormatter
from matplotlib.backends.backend_qt5agg \
    import (NavigationToolbar2QT as NavigationToolbar)
from matplotlib import cm
#Abrir guia
import subprocess
import cv2 as cv

myappid = 'com.UTP.OAUTP.CRT'
QtWin.setCurrentProcessExplicitAppUserModelID(myappid)

class MainWindow(QWidgets.QMainWindow, Ui_MainWindow):
    pass
    def __init__(self, *args, **kwargs):
        QMainWindow.__init__(self, *args, **kwargs)
        #Adquiere la hora de arranque de la GUI
        self.StartHour = str(time.strftime("%m-%d-%Y-%H-%M-%S"))
        #Desviaciones iniciales de Calibración y
        #el vector a almechar en excel
        self.VarAz = 0
        self.VarEl = 0
        self.number = 0
        self.Save = np.zeros((100, 5), dtype = object)
        self.Save[0][0] = 'desv. _Az'
        self.Save[0][1] = 'desv. _El'
        self.Save[0][2] = 'Voltaje'
        self.Save[0][3] = 'Hora_Inicio'

```

```

self.Save[0][4] = 'Hora_Final'
self.setupUi(self)
#Deshabilitar botón "Ver"
self.pushButton_7.setEnabled(False)
#Validación del ingreso de valores tipo double en
# las entradas de texto numéricas
self.lineEdit.setValidator(QtGui.QDoubleValidator())
self.lineEdit_2.setValidator(QtGui.QDoubleValidator())
self.lineEdit_6.setValidator(QtGui.QDoubleValidator())
self.lineEdit_7.setValidator(QtGui.QDoubleValidator())
self.lineEdit_11.setValidator(QtGui.QDoubleValidator())
self.lineEdit_12.setValidator(QtGui.QDoubleValidator())
self.lineEdit_23.setValidator(QtGui.QDoubleValidator())
self.lineEdit_24.setValidator(QtGui.QDoubleValidator())
self.lineEdit_31.setValidator(QtGui.QDoubleValidator())
#Validación del ingreso de valores tipo double en
# las entradas de texto numéricas enteras
self.Picsize.setValidator(QtGui.QIntValidator())
self.Picsize_2.setValidator(QtGui.QIntValidator())
self.lineEdit_30.setValidator(QtGui.QIntValidator())
#Widget de graficación 2d en fondo blanco
self.widget.setBackground('w')
self.widget_2.setBackground('w')
self.widget_3.setBackground('w')
#Deshabilitar botón Referencia y Tránsito
self.pushButton_2.setEnabled(False)
self.pushButton_3.setEnabled(False)
#Conecta botón "Cielo de fondo" a función plot1()
self.pushButton.clicked.connect(self.plot1)
#Conecta botón "Referencia" a función plot2()
self.pushButton_2.clicked.connect(self.plot2)
#Conecta botón "Tránsito" a función plot3()
self.pushButton_3.clicked.connect(self.plot3)
#Conecta botón "Ver" a función Gaussian()
self.pushButton_7.clicked.connect(self.Gaussian)
#Conecta botón "Enviar" de la ventana "Adquisición
# de datos" a función Send()

```

```

self.pushButton_4.clicked.connect(self.Send)
#Conecta botón "Enviar" de la ventana "Escaneo"
# a función Send()
self.pushButton_5.clicked.connect(self.SendC)
#Conecta botón "Enviar" de la ventana
# "Calibración" a función Send()
self.pushButton_6.clicked.connect(self.SendD)
self.Guia_0.triggered.connect(self.PDF)
self.AcercaDe.triggered.connect(self.Imagen)
self.pushButton_8.clicked.connect(self.Reinicio)
#Validación de coordenadas no vacías
self.lineEdit.textChanged.connect(self.disablebutton)
self.lineEdit_2.textChanged.connect(self.disablebutton)
self.lineEdit_6.textChanged.connect(self.disablebutton)
self.lineEdit_7.textChanged.connect(self.disablebutton)
self.lineEdit_11.textChanged.connect(self.disablebutton)
self.lineEdit_12.textChanged.connect(self.disablebutton)
self.lineEdit_23.textChanged.connect(self.disablebutton)
self.lineEdit_24.textChanged.connect(self.disablebutton)
self.lineEdit_30.textChanged.connect(self.disablebutton)
self.lineEdit_31.textChanged.connect(self.disablebutton)
self.Picsize.textChanged.connect(self.disablebutton)
self.Picsize_2.textChanged.connect(self.disablebutton)
self.guardado = str(os.path.expanduser("~\\Documents") \
+ "\\CRT")
#### CARPETAS PARA GUARDADO DE DATOS ####

```

```

try:

```

```

    os.mkdir(self.guardado)
    path_gauss = str(self.guardado + "\\Gauss")
    os.mkdir(path_gauss)
    path_esc = str(self.guardado + "\\Escaneo")
    os.mkdir(path_esc)
    path_cal = str(self.guardado + "\\Calibracion")
    os.mkdir(path_cal)
    path_gaussD = str(path_gauss + "\\Datos")
    os.mkdir(path_gaussD)
    path_escD = str(path_esc + "\\Datos")

```

```

os.mkdir(path_escD)
path_calD =str(path_cal + "\\Datos")
os.mkdir(path_calD)
path_gaussG = str(path_gauss + "\\Graficas")
os.mkdir(path_gaussG)
path_escG = str(path_esc + "\\Graficas")
os.mkdir(path_escG)
path_calG = str(path_cal + "\\Graficas")
os.mkdir(path_calG)
path_escF = str(path_esc + "\\Fotos")
os.mkdir(path_escF)
path_calD = str(path_cal + "\\Desviaciones")
os.mkdir(path_calD)
except:
    pass
### VERIFICACIÓN DE ENTRADAS DE TEXTO PARA ACTIVAR BOTONES ###
def disablebutton(self):
    if len(self.lineEdit_23.text())>0 and \
        len(self.lineEdit_24.text())>0 and \
        len(self.lineEdit_30.text())>0 and \
        len(self.lineEdit_31.text())>0:
        if self.pushButton_2.isEnabled() or \
            self.pushButton.isEnabled():
            self.pushButton_3.setDisabled(True)
        else:
            self.pushButton_3.setDisabled(False)
    else:
        self.pushButton_3.setDisabled(True)

    if len(self.lineEdit.text()) > 0 and \
        len(self.lineEdit_2.text())>0:
        self.pushButton_4.setDisabled(False)
    else:
        self.pushButton_4.setDisabled(True)

    if len(self.lineEdit_6.text()) > 0 and \
        len(self.lineEdit_7.text()) > 0 and \

```

```

        len(self.Picsize.text()) >0 and \
        len(self.Picsize_2.text()) > 0:
            self.pushButton_5.setDisabled(False)
    else:
        self.pushButton_5.setDisabled(True)

    if len(self.lineEdit_11.text()) > 0 \
        and len(self.lineEdit_12.text())>0:
        self.pushButton_6.setDisabled(False)
    else:
        self.pushButton_6.setDisabled(True)

#### GUIA DE USUARIO ###
def PDF (self):
    path = "Guia.pdf"
    os.startfile(path)

#### ACERCA DE ###
def Imagen(self):
    path = "Acercade.png"
    image = Image.open(path)
    image.show()

### REINICIO CALIBRACION ###
def Reinicio(self):
    self.StartHour = str(time.strftime("%m-%d-%Y-%H-%M-%S"))
    self.VarAz = 0
    self.VarEl = 0
    self.number = 0
    self.widget_4.canvas.axes.clear()

#####FUNCIÓN DAQ() PARA LA ADQUISICIÓN DE DATOS#####
def ADQ (self , time):
    with nidaqmx.Task() as task:
        #Tiempo de adquisición
        Time = time
        #Frecuencia de muestreo

```

```

Adq_time = int(self.Picsize_2.text())
#Número de muestras
NSAMPLES = int(Adq_time*Time)
#Canal ai0 para la lectura de datos
task.ai_channels.add_ai_voltage_chan("myDAQ1/ai0")
#Configuración de la DAQ para la adquisición
task.timing.cfg_samp_clk_timing(Adq_time, source="", \
    active_edge=Edge.RISING, \
    sample_mode=AcquisitionType.FINITE, \
    samps_per_chan=NSAMPLES)#Tiempo de muestreo
#Inicia la adquisición de datos
task.start()
Read = np.array(task.read(NSAMPLES, \
    timeout= Time))*1e6
task.stop()
#Voltaje de salida en dBuV
Read = np.absolute(Read)
Read = 20*np.log10(Read)
if int(Time) > 6:
    #Diseño del filtro pasa bajos
    def butter_lowpass(cutoff, fs, order=5):
        nyq = 0.5 * fs
        normal_cutoff = cutoff / nyq
        b, a = butter(order, normal_cutoff, \
            btype='low', analog=False)
        return b, a
    #Aplicación con del filtro con el diseño implementado
    def butter_lowpass_filter(data, cutoff, fs, order=5):
        b, a = butter_lowpass(cutoff, fs, order=order)
        y = filtfilt(b, a, data, axis = 0)
        return y
    #Filtro
    order = 2
    fs = Adq_time
    cutoff = 0.5
    #Devuelve datos filtrados para Tiempo mayores a 5 S
    y = butter_lowpass_filter(Read, cutoff, fs, order)

```

```

        else :
            #Devuelve datos sin filtrar para Tiempos menores a 5 S
            y = Read
        return y

### ADQUISICIÓN DE LA POSICIÓN ACTUAL DE LA ANTENA ###
def Position(self):
    #Comando para preguntar la posición actual
    #en Altitud en Pstrotator
    POSEL = '<PST>EL?</PST>'
    POSEL = POSEL.encode()
    #Comando para preguntar la posición actual
    #en Acimut en Pstrotator
    POSAZ = '<PST>AZ?</PST>'
    POSAZ = POSAZ.encode()
    #Creación del Socket de envío
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    #Creación del Socket de recepción
    sock_2 = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    #Vinculación del Socket de recepción
    # con el puerto UDP 12001 e IP local
    sock_2.bind(('', 12001))
    #Reporte de posición el Altitud
    sock.sendto(POSEL, ('localhost', 12000))
    EL, add = sock_2.recvfrom(1024)
    #Reporte de posición en Acimut
    sock.sendto(POSAZ, ('localhost', 12000))
    AZ, add = sock_2.recvfrom(1024)
    #Cierre y finalización de la conexión
    sock.close()
    sock_2.close()
    #Decodificación del valor recibido
    EL = EL.decode()
    AZ = AZ.decode()
    EL = float(EL[3:])
    AZ = float(AZ[3:])
    #Reporte de posición en Azimut

```

```

    #y Elevación actual
    return EL, AZ

### ENVÍO DE COORDENADAS A PSTROTATOR ###
def SendPosition(self, AZ, EL):
    #Verificación de coordenadas
    DecAZ, DecEL = round(10*(AZ-int(AZ))), round(10*(EL-int(EL)))
    ModAZ, ModEL = DecAZ % 2, DecEL % 2
    if 0<=AZ<=360 and 0<=EL<=90.1 and ModAZ == 0 and ModEL == 0:
        #Chequeo de la posición actual
        AC_EL, AC_AZ = self.Position()
        if int(10*(AZ - 0.1)) <= int(10*AC_AZ) \
           <= int(10*(AZ + 0.1)) and int(10*(EL - 0.1)) <= \
           int(10*AC_EL) <= int(10*(EL + 0.1)):
            pass
        #Envío de las coordenadas
        else:
            #Socket de envío
            sock = socket.socket(socket.AF_INET, \
                                socket.SOCK_DGRAM)
            #Socket de recepción
            sock_2 = socket.socket(socket.AF_INET, \
                                   socket.SOCK_DGRAM)
            #Vinculación del Socket de recepción con
            #el puerto UDP 12001 e IP local
            sock_2.bind(('', 12001))
            #Coordenada codificada
            POS='<PST><ELEVATION>'+ str(EL) + \
                '</ELEVATION><AZIMUTH>'+ str(AZ) + \
                '</AZIMUTH></PST>'
            POS=POS.encode()
            #Envío de las coordenadas
            sock.sendto(POS, ('localhost', 12000))
            #Escala de Acimut y Altitud
            AZ = int(10*AZ)
            EL = int(10*EL)
            i, j = 0, 0

```



```

u, v = 0, 0
while(True):
    #Escuchar el reporte durante el movimiento
    Pos, add = sock_2.recvfrom(1024)
    PosD = int(10*float(Pos[3:]))
    #Para movimiento el Altitud
    if Pos[0] == 69:
        i = 1
        if (EL - 2) <= PosD <= (EL + 2):
            sock.sendto(POS, ('localhost',12000))
            if (EL - 1) <= EL <= (EL + 1):
                u = 1
    #Para movimiento en Acimut
    if Pos[0] == 65:
        j = 1
        if (AZ - 2) <= PosD <= (AZ + 2):
            sock.sendto(POS, ('localhost',12000))
            if (AZ - 1) <= PosD <= (AZ + 1):
                v = 1
    #Chequeo si la montura llegó al punto final
    if i*j == 0:
        if u == 1 or v == 1:
            break
    else:
        if u == 1 and v == 1:
            break
    #Cierre de la conexión
    sock_2.close()
    sock.close()
else:
    #Mensaje de advertencia para coordenadas erroneas
    QMessageBox.about(self, "Advertencia", \
        "Las coordenadas ingresadas están fuera del rango" \
        "son de número decimal impar")

##### PARA LA VENTANA "ADQUISICIÓN DE DATOS" #####
### Botón "Cielo de fondo" ###

```

```

def plot1(self):
    self.Picsize_2.setText('10')
    self.widget.clear()
    Time = 5
    #Adquisición durante 5 S
    Read = self.ADQ(Time)
    #Limites y ejes de la gráfica
    pen = pg.mkPen(color=(255, 0, 0))
    styles = {'color': 'r', 'font-size': '12px'}
    self.widget.setLabel('left', 'Respuesta_(dBuV)', **styles)
    self.widget.setLabel('bottom', 'Tiempo_(S)', **styles)
    x = np.linspace(0, len(Read), len(Read))/Time
    #Grafica de los datos obtenidos en dBuV
    self.widget.plot(x, Read, pen=pen)
    #Habilita "Referencia"
    self.pushButton_2.setEnabled(True)
    #Deshabilitar el botón actual y botón "Ver"
    self.pushButton.setEnabled(False)
    self.pushButton_7.setEnabled(False)
    #Guardar valor promedio en voltios
    self.Fondo = 10**((np.mean(Read)/20)/(1e6))
    #Actualización de repuesta en voltios
    self.lineEdit_27.setText(str(round(self.Fondo,5)) + '_V')

```

Botón "Referencia"

```

def plot2(self):
    self.Picsize_2.setText('10')
    self.widget_2.clear()
    #Adquisición durante 5 S
    Time = 5
    Read = self.ADQ(Time)
    #Limites y ejes de la gráfica
    pen = pg.mkPen(color=(255, 0, 0))
    x = np.linspace(0, len(Read), len(Read))/Time
    styles = {'color': 'r', 'font-size': '12px'}
    self.widget_2.setLabel('left', 'Respuesta_(dBuV)', **styles)
    self.widget_2.setLabel('bottom', 'Tiempo_(S)', **styles)

```

```

#Grafica de los datos obtenidos en dBuV
self.widget_2.plot(x, Read, pen=pen)
#Deshabilitar botón actual
self.pushButton_2.setEnabled(False)
#Habilita el botón tránsito
self.pushButton_3.setEnabled(True)
#Almacena el valor promedio en voltios
self.Cupula = 10**((np.mean(Read)/20)/(1e6))
#Actualización de respuesta en voltios
self.lineEdit_28.setText(str(round(self.Cupula,5)) + '_V')

#### Botón "Tránsito" ####
def plot3(self):
    self.Picsize_2.setText('10')
    self.widget_3.clear()
    #Tiempo de adquisición ingresado
    Time = int(self.lineEdit_30.text())*60
    #Mensaje de espera durante la adquisición
    Wait = QMessageBox()
    Wait.setIcon(QMessageBox.Information)
    Wait.setWindowTitle("Mensaje")
    Wait.setText("'La adquisición está en curso.' \
        'Este proceso puede tardar varios minutos'")
    Wait.setWindowModality(QtCore.Qt.NonModal)
    Wait.show()
    QtCore.QCoreApplication.processEvents()
    #Adquisición durante el tiempo ingresado
    Read = self.ADQ(Time)
    #Eje x definido en tiempo
    x = np.linspace(0, Time, len(Read))
    #Almacenamiento de los datos obtenidos en archivo excel
    self.Gauss_1 = Read
    self.Save_1 = str(time.strftime("%m_%d_%Y_%H_%M_%S"))
    pathname = self.guardado + "\\Gauss\\Datos\\"
    df = pd.DataFrame(Read)
    out_path = pathname + self.Save_1 + ".xlsx"
    writer = pd.ExcelWriter(out_path, engine='xlsxwriter')

```

```

df.to_excel(writer, sheet_name='Sheet1', \
            index = False, header = False)
writer.save()
#Lectura de temperatura de referencia y
#Declinación del objeto
Tcu = float(self.lineEdit_23.text())
Dec = float(self.lineEdit_24.text())
#Limites y ejes de la gráfica
pen = pg.mkPen(color=(255, 0, 0))
styles = {'color': 'r', 'font-size': '12px'}
self.widget_3.setLabel('left', 'Respuesta_(dBuV)', **styles)
self.widget_3.setLabel('bottom', 'Tiempo_(S)', **styles)
#Grafica de los datos obtenidos en dBuV
self.widget_3.plot(x, Read, pen=pen)
#Deshabilitar el botón tránsito
self.pushButton_3.setEnabled(False)
#Habilitar el botón "Ver" y "Cielo de Fondo"
self.pushButton_7.setEnabled(True)
self.pushButton.setEnabled(True)
#Lectura de tamaño del objeto,
size = float(self.lineEdit_31.text())
#Respuesta al cielo de fondo
Fondo = float(self.Fondo)
#Respuesta al apuntar la cúpula
Cupula = float(self.Cupula)
#Datos tránsito en voltios
Read = Read/20
Read = 10**((Read)/(1e6))
#Constante de proporcionalidad K/V
c = (20 - Tcu)/(Fondo - Cupula)
#Temperatura del sistema
Tsys = c*Cupula-Tcu
Tsol = c*round(np.max(Read),5) - Tsys - 20
y = self.Gauss_1
#Función gaussiana a optimizar
def gauss(x, H, A, x0, sigma):
    return H + A * np.exp(-(x - x0) ** 2 / (2 * sigma ** 2))

```

```

#Optimización del ajuste Gaussiano
def gauss_fit(x, y):
    mean = sum(x * y) / sum(y)
    sigma = np.sqrt(sum(y * (x - mean) ** 2) / sum(y))
    popt, pcov = optimize.curve_fit(gauss, \
        x, y, p0=[min(y), max(y), mean, sigma])
    return popt
#Constantes de la función gaussiana
try:
    H, A, x0, sigma = gauss_fit(x, y)
    Yg = gauss(x, *gauss_fit(x, y))
except:
    H, A, x0, sigma = 0, 0, 0, 0
    Yg = y
#HPBW en segundos
Posg = round((np.sqrt(2*np.log(2))*sigma),3)
#HPBW en grados
HPBW = round((Posg)/240*np.cos(Dec*3.1416/180), 3)
#HPBW de convolución
HPBW = round(np.sqrt(HPBW**2 - np.log(2)/2*size**2), 3)
#Areas angulares
AreaAnt = 3.14159*((2*HPBW) ** 2)
AreaObj = 3.14159*(size ** 2)
#Area efectiva de la antena y eficiencia
#Temperatura de brillo
Tb = AreaAnt/AreaObj*Tsol
#Finalización de la sesión
Wait.close()
#Salida de resultados
self.lineEdit_5.setText(str(Posg) + 's')
self.lineEdit_22.setText(str(HPBW) + '°')
self.lineEdit_29.setText(str(round(np.max(Read),5)) + 'V')
self.lineEdit_10.setText(str(round(Tsys, 3)))
self.lineEdit_21.setText(str(round(Tsol, 3)))
self.lineEdit_32.setText(str(round(Tb, 3)))

```

Botón "Ver"

```

def Gaussian(self):
    self.Picsize_2.setText('10')
    #Lectura de los parámetros ingresados
    Dec = float(self.lineEdit_24.text())
    size = float(self.lineEdit_31.text())
    y = self.Gauss_1
    Time = int(self.lineEdit_30.text())*60
    #Eje x en tiempo
    x = np.linspace(0, Time, len(y))
    #Función Guassiana a optimizar
    def gauss(x, H, A, x0, sigma):
        return H + A * np.exp(-(x - x0) ** 2 / (2 * sigma ** 2))
    #Optimización del ajuste Gaussiano
    def gauss_fit(x, y):
        mean = sum(x * y) / sum(y)
        sigma = np.sqrt(sum(y * (x - mean) ** 2) / sum(y))
        popt, pcov = optimize.curve_fit(gauss, \
            x, y, p0=[min(y), max(y), mean, sigma])
        return popt
    try:
        #Constantes de la función Gaussiana
        H, A, x0, sigma = gauss_fit(x, y)
        #Arreglo con ajuste gaussiano más aproximado
        Yg = gauss(x, *gauss_fit(x, y))
    except:
        H, A, x0, sigma = 0, 0, 0, 0
        Yg = np.zeros(len(y))
    #Calculo del HPBW
    Posg = round((np.sqrt(2*np.log(2))*sigma), 3)
    HPBW = round((Posg)/240*np.cos(Dec*3.1416/180), 3)
    HPBWG = str(round(np.sqrt(HPBW**2 - np.log(2)/2*size**2), 3))
    #Gráficas de los datos y ajuste Gaussiano
    fig = plt.figure(1)
    ax = fig.add_subplot(1, 1, 1)
    plt.plot(x, y, label = str(abs(Posg)) + '_s')
    plt.plot(x, Yg, color = 'r')
    ax.axvline(x = Posg + x0, color = 'r', \

```

```

        label = str(HPBWG) + ' ° '
#Ajuste de ejes
ax.set_xlabel('Tiempo (S)')
ax.set_ylabel('Salida (dBuV)')
plt.legend(loc = 'best')
#Mostrar
plt.show()
#Guardar imagen
pathname = self.guardado + "\\Gauss\\Graficas\\"
fig.savefig(pathname + self.Save_1 + ".png")

#### Botón "Envío" #####
def Send(self):
    self.Picsize_2.setText('10')
    #Coordenadas ingresadas
    EL=float(self.lineEdit.text())
    AZ=float(self.lineEdit_2.text())
    #Envío
    self.SendPosition(AZ, EL)
    #Actualización de posición final
    AC_EL, AC_AZ = self.Position()
    self.lineEdit_3.setText(str(AC_EL))
    self.lineEdit_4.setText(str(AC_AZ))
    self.lineEdit_8.setText(str(AC_EL))
    self.lineEdit_9.setText(str(AC_AZ))
    self.lineEdit_13.setText(str(AC_EL))
    self.lineEdit_14.setText(str(AC_AZ))

##### PARA LA VENTANA "ESCANEEO" #####
#### Botón "Envío" ####
def SendC(self):
    self.widget_6.canvas.axes.clear()
    #Coordenadas ingresadas
    EL=float(self.lineEdit_6.text())
    AZ=float(self.lineEdit_7.text())
    #Tamaño de la imagen
    Pic_Size = int(self.Picsize.text())

```

```

#Frecuencia de muestreo
Adq_time = int(self.Picsize_2.text())
#Verificación de coordenadas
DecAZ, DecEL = round(10*(AZ-int(AZ))), round(10*(EL-int(EL)))
ModAZ, ModEL = DecAZ % 2, DecEL % 2
#Tamaño del eje
Size = int(self.comboBox_3.currentText())/2
#Eje y para valores de tiempo
y = [None]*(Pic_Size+1)
plt.clf()
plt.figure(figsize=(7.7,4.8))
#Validación de las coordenadas
if (AZ - Size) < 360 and (AZ + Size) > 0 and \
    (EL - Size) > 5 and (EL + Size) < 90 and \
    ModAZ == 0 and ModEL == 0 and Adq_time <= 300:
    #Mensaje de espera durante la sesión
    Wait = QMessageBox()
    Wait.setIcon(QMessageBox.Information)
    Wait.setWindowTitle("Mensaje")
    Wait.setText("'La adquisición está en curso.' \
        "'Este proceso puede tardar varios minutos'")
    Wait.setWindowModality(QtCore.Qt.NonModal)
    Wait.show()
    QtCore.QCoreApplication.processEvents()
    #Envío al punto central
    self.SendPosition(AZ, EL)
    #Hora de inicio
    Init = str(time.strftime("%m-%d-%Y-%H-%M-%S"))
    self.lineEdit_20.setText(str(time.strftime("%H:%M:%S")))
    #Socket de envío
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    #Para Acimut constante
    if 0 <= AZ <= 30 or 330 <= AZ <= 360 or 150 <= AZ <= 210:
        self.widget_6.canvas.axes.set_xlabel('Elevación')
        label = 'Elevación'
    #Mínimo y máximo de altitud
    Min = EL - Size

```



```

Max = EL + Size
#Envío a acimut y mínimo altitud
self.SendPosition(AZ, Min)
#Tiempo para magnitud de movimiento escogido
start = time.time()
self.SendPosition(AZ, Max)
Time_1 = float(time.time() - start)
start = time.time()
self.SendPosition(AZ, Min)
Time_2 = float(time.time() - start)
Time = (Time_1 + Time_2)/2
Time = round(Time + Time*0.1, 1)
#Numero de muestras
NSAMPLES = int(Adq_time*Time)
#Imagen a llenar
Datos = np.zeros((Pic_Size+1, NSAMPLES))
I = 2
Ref = 0
#ADQUISICIÓN DE DATOS
while Ref <= Pic_Size:
    Mod = I % 2
    #Si el movimiento será ascendente
    if Mod == 0:
        #Hora de inicio
        y[Ref] = str(time.strftime("%H:%M:%S"))
        #Envío a la posición máxima
        POS='<PST><ELEVATION>' + str(Max) + \
            '</ELEVATION></PST>'
        POS=POS.encode()
        sock.sendto(POS, ('localhost',12000))
        #Inicia la adquisición
        Read = self.ADQ(Time)
        #Almacenamiento de datos en la imagen
        for i in range(0, NSAMPLES):
            Datos[Ref][i] = Read[i]
        Ref = Ref + 1
    #Si el movimiento será descendente

```

```

else:
    #Hora de inicio
    y[Ref] = str(time.strftime("%H:%M:%S"))
    #Envío a la posición mínima
    POS='<PST><ELEVATION>' + str(Min) + \
        '</ELEVATION></PST>'
    POS=POS.encode()
    sock.sendto(POS, ('localhost',12000))
    #Inicia la adquisición
    Read = self.ADQ(Time)
    #Almacenamiento de datos en la imagen
    for i in range(0, NSAMPLES):
        Datos[Ref][(NSAMPLES - 1) - i]=Read[i]
        Ref = Ref + 1
    I = I + 1
    #Eje horizontal de la matriz final
    x = np.linspace((EL - Size), (EL + Size), NSAMPLES+1)
#Para elevación constante
else:
    self.widget_6.canvas.axes.set_xlabel('Azimut')
    label = 'Azimut'
    #Mínimo y máximo de altitud
    Min = AZ - Size
    Max = AZ + Size
    #Envío a altitud y mínimo de acimut
    self.SendPosition(Min, EL)
    #Tiempo para magnitud de movimiento escogido
    start = time.time()
    self.SendPosition(Max, EL)
    Time_1 = float(time.time() - start)
    start = time.time()
    self.SendPosition(Min, EL)
    Time_2 = float(time.time() - start)
    Time = (Time_1 + Time_2)/2
    Time = round(Time + Time*0.1, 1)
    #Numero de muestras
    NSAMPLES = int(Time*Adq_time)

```

```

#Imagen a llenar
Datos = np.zeros(( Pic_Size+1, NSAMPLES))
I = 2
Ref = 0
#ADQUISICIÓN DE DATOS
while Ref <= Pic_Size:
    Mod = I % 2
    #Si el movimiento será ascendente
    if Mod == 0:
        #Hora de inicio
        y[Ref] = str(time.strftime("%H:%M:%S"))
        #Envío a la posición máxima
        POS='<PST><AZIMUTH>' + str(Max) + \
            '</AZIMUTH></PST>'
        POS=POS.encode()
        sock.sendto(POS, ('localhost',12000))
        #Adquisición de datos
        Read = self.ADQ(Time)
        #Almacenamiento de datos en la imagen
        for i in range(0, NSAMPLES):
            Datos[Ref][i] = Read[i]
        Ref = Ref + 1
    #Si el movimiento será descendente
    else:
        #Hora de inicio
        y[Ref] = str(time.strftime("%H:%M:%S"))
        #Envío a la posición mínima
        POS='<PST><AZIMUTH>' + str(Min) + \
            '</AZIMUTH></PST>'
        POS=POS.encode()
        sock.sendto(POS, ('localhost',12000))
        #Adquisición de datos
        Read = self.ADQ(Time)
        #Almacenamiento de datos en la imagen
        for i in range(0, NSAMPLES):
            Datos[Ref][(NSAMPLES -1 - i)]=Read[i]
        Ref = Ref +1

```

```

        I = I + 1
        #Eje horizontal de la matriz final
        x = np.linspace((AZ - Size), (AZ + Size), NSAMPLES+1)
#Cierre y finalización de la comunicación
        sock.close()
#Eje vertical de la matriz final
        u = np.arange(0, Ref)
#Hora de finalización
        self.lineEdit_19.setText(str(time.strftime("%H:%M:%S")))
#Actualización de la posición actual
        AC_EL, AC_AZ = self.Position()
        self.lineEdit_3.setText(str(AC_EL))
        self.lineEdit_4.setText(str(AC_AZ))
        self.lineEdit_8.setText(str(AC_EL))
        self.lineEdit_9.setText(str(AC_AZ))
        self.lineEdit_13.setText(str(AC_EL))
        self.lineEdit_14.setText(str(AC_AZ))
#Almacenamiento de la matriz resultante
#en archivo excel
        z = Datos
        pathname = self.guardado + "\\Escaneo\\Datos\\"
        df = pd.DataFrame(Datos)
        out_path = pathname + Init + ".xlsx"
        writer = pd.ExcelWriter(out_path, engine='xlsxwriter')
        df.to_excel(writer, sheet_name='Sheet1', \
                    index = False, header = False)
        writer.save()
#Normalización de los datos
        Min_Value = np.min(Datos)
        Max_Value = np.max(Datos)
        Datos = (Datos - Min_Value)/(Max_Value - Min_Value)
        size = np.shape(Datos)
        Datosn = np.zeros(size)
        for i in range (0, size[0]):
            Datosn[i][:] = Datos[size[0]-1-i][:]
#Creación de la imagen en blanco y negro
        Imagen = Image.fromarray(np.uint8(255*Datosn), mode = 'L')

```

```

#Almacenamiento del archivo .jpg
pathname = self.guardado + "\\Escaneo\\Fotos\\"
Imagen.save(pathname + Init + ".jpg")
#Lectura de la imagen
self.Grafica.setPixmap(QtGui.QPixmap(str(pathname + \
    Init + ".jpg")))
#Almacenamiento de la gráfica en mapa de colores
#en archivo .jpg
fig = plt.figure(2)
ax2 = fig.add_subplot(1, 1, 1)
#Eje vertical
ax2.set_ylabel('Tiempo')
ax2.set_yticks(u, y)
ax2.set_xlabel(label)
#Superficie mapeada
surf = ax2.pcolormesh(x, y, z, \
    cmap = cm.get_cmap("hot"), antialiased=True)
#Barra de colores
fig.colorbar(surf, ax = ax2)
#Almacenamiento
pathname = self.guardado + "\\Escaneo\\Graficas\\"
plt.savefig(pathname + Init + ".png")
plt.close(2)
#Display de la gráfica en mapa de colores:
#Eje vertical
self.widget_6.canvas.axes.set_yticklabels(y, \
    fontsize = 8)
self.widget_6.canvas.axes.set_ylabel('Tiempo')
#Superficie mapeada
surf = self.widget_6.canvas.axes.pcolormesh(x, y, z, \
    cmap = cm.get_cmap("hot"), antialiased=True)
#Barra de colores
cb = self.widget_6.fig.colorbar(surf, \
    cax = self.widget_6.fig.add_axes([0.75, 0.5, 0.03, 0.38]))
#Mostrar
self.widget_6.canvas.draw()
cb.remove()

```

```

        #Cierre de ventana de espera
        Wait.close()
    else:
        #Mensaje de advertencia para coordenadas erroneas
        QMessageBox.about(self, "Advertencia", \
            "Las_coordenadas_ingresadas_están_fuera_del_rango_o" \
            "son_de_número_decimal_impar._Sample_mayor_a_300")

##### PARA LA VENTANA "Calibración" #####
### Botón "Envío" ###
def SendD(self):
    self.Picsize_2.setText('10')
    #Coordenadas ingresadas
    EL=round(float(self.lineEdit_11.text()), 1)
    AZ=round(float(self.lineEdit_12.text()), 1)
    #Validación del punto decimal
    DecAZ, DecEL = round(10*(AZ-int(AZ))), round(10*(EL-int(EL)))
    ModAZ, ModEL = DecAZ % 2, DecEL % 2
    #Ancho de la ventana
    Anchor = int(self.comboBox.currentText())
    #Validación de coordenadas
    if (AZ - Anchor/2) < 360 and (AZ + Anchor/2) > 0 and \
        (EL - Anchor/2) > 5 and (EL + Anchor/2) < 90 and \
        ModAZ == 0 and ModEL == 0:
        #Mensaje de espera
        Wait = QMessageBox()
        Wait.setIcon(QMessageBox.Information)
        Wait.setWindowTitle("Mensaje")
        Wait.setText("'La_adquisición_está_en_curso'." \
            "Este_proceso_puede_tardar_varios_minutos'")
        Wait.setWindowModality(QtCore.Qt.NonModal)
        Wait.show()
        QtCore.QCoreApplication.processEvents()
        #Resolución de la ventana
        Step = float(self.comboBox_2.currentText())
        Up = -1
        I = 2

```

```

#Tiempo de adquisición por dato y
#Frecuencia de muestreo
Time = 0.5
Adq_time = 10
#Matriz de datos a llenar
Datos = np.zeros((int(Anchor/Step + 1), \
    int(Anchor/Step + 1)))
#Hora de inicio
Init = str(time.strftime("%m-%d-%Y-%H-%M-%S"))
self.lineEdit_15.setText(str(time.strftime("%H:%M:%S")))
#Envío a la posición central
self.SendPosition(AZ, EL)
#Adquisición de datos
Reada = self.ADQ(Time)
Center = np.mean(Reada)
#Coordenadas mínimas y máximas de la matriz
MinAZ = round(AZ - Anchor/2, 1)
MaxAZ = round(AZ + Anchor/2, 1)
MaxEL = round(EL + Anchor/2, 1)
MinEL = round(EL - Anchor/2, 1)
#Eje horizontal y vertical
X = np.arange((MinAZ - Step/2), (MaxAZ + Step), Step)
Y = np.arange((MinEL - Step/2), (MaxEL + Step), Step)
i = 0
#Para adquisición ascendente
if AZ >= 180:
    #Envío a posición mínima
    self.SendPosition(MinAZ, MinEL)
    #Inicio de recorrido
    while int(10*MinEL) <= int(10*MaxEL):
        j = 0
        Mod = i%2
        #Si es en acimut ascendente
        if Mod == 0:
            while int(10*MinAZ) <= int(10*MaxAZ):
                #Saltar posición central
                if int(10*MinAZ) == int(10*AZ) and \

```

```

        int(MinEL*10) == int(10*EL):
        MinAZ = MinAZ + Step
        Datos[i][j] = Center
        j = j + 1
        #Envío del punto coordenado
        self.SendPosition(MinAZ, MinEL)
        #Adquisición de datos
        Read = self.ADQ(Time)
        #Almacenamiento en la matriz
        Datos[i][j] = np.mean(Read)
        #Comparación valor máximo y desviaciones
        if Datos[i][j] > Up:
            Up = Datos[i][j]
            xs = MinAZ - AZ
            ys = MinEL - EL
        j = j + 1
        #Siguiente posicion en acimut
        MinAZ = round(MinAZ + Step, 1)
        #Reconstruye acimut mínimo
        MinAZ = round(AZ - Anchor/2, 1)
        #Si es en acimut descendente
    else:
        while int(MaxAZ*10) >= int(MinAZ*10):
            #Saltar la posición central
            if int(MaxAZ*10) == int(AZ*10) and \
                int(MinEL*10) == int(10*EL):
                MaxAZ = MaxAZ - Step
                Datos[i][int(Anchor/Step)-j] = Center
                j = j + 1
            #Envío al punto coordenado
            self.SendPosition(MaxAZ, MinEL)
            #Adquisición de datos
            Read = self.ADQ(Time)
            #Almacenamiento en la matriz
            Datos[i][int(Anchor/Step)-j] = np.mean(Read)
            #Comparación valor máximo y desviaciones
            if Datos[i][int(Anchor/Step)-j] > Up:

```



```

        Up = Datos[ i ][ int( Anchor/Step)-j ]
        xs = MaxAZ - AZ
        ys = MinEL - EL
        j = j + 1
        #Siguiente posición en acimut
        MaxAZ = round(MaxAZ - Step, 1)
        #Reconstruye acimut máximo
        MaxAZ = round(AZ + Anchor/2, 1)
        i = i + 1
        I = I + 1
        #Siguiente posición en altitud
        MinEL = round(MinEL + Step, 1)
#Para adquisición descendente
    else :
        MinEL = MaxEL
        #Envío a la posición máxima de altitud
        #y mínima de acimut
        self.SendPosition(MinAZ, MinEL)
        #Inicio del recorrido
        while int(10*MinEL) >= int(10*round(EL - Anchor/2, 1)):
            j = 0
            Mod = I%2
            #Si es en acimut ascendente
            if Mod == 0:
                while int(10*MinAZ) <= int(10*MaxAZ):
                    #Saltar la posición central
                    if int(10*MinAZ) == int(10*AZ) and \
                        int(MinEL*10) == int(10*EL):
                        MinAZ = MinAZ + Step
                        Datos[ int( Anchor/Step)-i ][ j ] = Center
                        j = j + 1
                    #Envío al punto coordenado
                    self.SendPosition(MinAZ, MinEL)
                #Adquisición de datos
                Read = self.ADQ(Time)
                #Almacenamiento en la matriz
                Datos[ int( Anchor/Step)-i ][ j ] = np.mean(Read)

```

```

#Comparación valor máximo y desviaciones
if Datos[int(Anchor/Step)-i][j] > Up:
    Up = Datos[int(Anchor/Step)-i][j]
    xs = MinAZ - AZ
    ys = MinEL - EL
    j = j + 1
#Siguiente posición en acimut
    MinAZ = round(MinAZ + Step, 1)
#Reconstruye el acimut mínimo
    MinAZ = round(AZ - Anchor/2, 1)
#Si es en acimut descendente
else:
    while int(MaxAZ*10) >= int(MinAZ*10):
        #Saltar la posición central
        if int(MaxAZ*10) == int(AZ*10) and \
            int(MinEL*10) == int(10*EL):
            MaxAZ = MaxAZ - Step
            Datos[int(Anchor/Step) - i]\
                [int(Anchor/Step)-j] = Center
            j = j + 1
        #Envío al punto coordenado
        self.SendPosition(MaxAZ, MinEL)
        #Adquisición de datos
        Read = self.ADQ(Time)
        #Almacenamiento en la matriz
        Datos[int(Anchor/Step)-i]\
            [int(Anchor/Step)-j] = np.mean(Read)
        #Comparación valor máximo y desviaciones
        if Datos[int(Anchor/Step)-i]\
            [int(Anchor/Step)-j] > Up:
            Up = Datos[int(Anchor/Step)-i]\
                [int(Anchor/Step)-j]
            xs = MaxAZ - AZ
            ys = MinEL - EL
            j = j + 1
        #Siguiente posición en acimut
        MaxAZ = round(MaxAZ - Step, 1)

```

```

        #Reconstruye acimut máximo
        MaxAZ = round(AZ + Anchor/2, 1)
        i = i + 1
        I = I + 1
        #Siguiente posición en altitud
        MinEL = round(MinEL - Step, 1)
#Actualiza la posición actual
        AC_EL, AC_AZ = self.Position()
        self.lineEdit_3.setText(str(AC_EL))
        self.lineEdit_4.setText(str(AC_AZ))
        self.lineEdit_8.setText(str(AC_EL))
        self.lineEdit_9.setText(str(AC_AZ))
        self.lineEdit_13.setText(str(AC_EL))
        self.lineEdit_14.setText(str(AC_AZ))
#Hora de finalización
        Fin = str(time.strftime("%H:%M:%S"))
        self.lineEdit_16.setText(Fin)
#Almacenamiento de los datos en archivo excel
        pathname = self.guardado + "\\Calibracion\\Datos\\"
        df = pd.DataFrame(Datos)
        out_path = pathname + Init + ".xlsx"
        writer = pd.ExcelWriter(out_path, engine='xlsxwriter')
        df.to_excel(writer, sheet_name='Sheet1', \
                    index = False, header = False)
        writer.save()
        Datos = np.absolute(Datos)
        Max_Value = np.max(Datos)
        plt.clf()
#Almacenamiento de la gráfica
#en mapa de colores en archivo .png
        fig3 = plt.figure(3)
        ax3 = fig3.add_subplot(1, 1, 1)
#Mapeo de color
        surf = ax3.pcolormesh(X, Y, Datos, \
                             cmap = cm.get_cmap("hot"))
#Barra de color
        fig3.colorbar(surf, ax = ax3)

```

```

#Nombre de ejes
ax3.set_xlabel('Azimut_(°)')
ax3.set_ylabel('Elevación_(°)')
#Almacenamiento en archivo .png
pathname = self.guardado + "\\Calibracion\\Graficas\\"
plt.savefig(pathname + Init + ".png")
plt.close(3)
#Display de la gráfica en mapa de colores:
self.widget_5.canvas.axes.clear()
#Superficie mapeada
Surf = self.widget_5.canvas.axes.pcolormesh(X, Y, Datos, \
      cmap = cm.get_cmap("hot"))
#Barra de colores
cb = self.widget_5.fig.colorbar(mappable = Surf, \
      cax = self.widget_5.fig.add_axes([0.78, 0.5, 0.03, 0.38]))
#Nombre de ejes
self.widget_5.canvas.axes.set_xlabel('Azimut_(°)')
self.widget_5.canvas.axes.set_ylabel('Elevación_(°)')
self.widget_5.canvas.draw()
#Ejes gráfica desviaciones
self.widget_4.canvas.axes.set_xlabel('Desv._Azimut')
self.widget_4.canvas.axes.set_ylabel('Desv._Elevacion')
#Magnitud desviación
self.widget_4.canvas.axes.scatter(xs,ys, marker = 'o', c = 'b')
#Almacenamiento de desviaciones por sesion
#valor máximo y horas
self.VarAz = self.VarAz + xs
self.VarEl = self.VarEl + ys
self.number = self.number + 1
self.Save[self.number][0] = xs
self.Save[self.number][1] = ys
self.Save[self.number][2] = Up
self.Save[self.number][3] = Init
self.Save[self.number][4] = Fin
pathname = self.guardado + "\\Calibracion\\Desviaciones\\"
df = pd.DataFrame(self.Save)
out_path = pathname + self.StartHour + ".xlsx"

```

```

writer = pd.ExcelWriter(out_path , engine='xlsxwriter')
df.to_excel(writer , sheet_name='Sheet1' , \
            index = False , header = False)
writer.save()
#Ejes mapa de desviaciones
majorx = np.arange(-6, 7, 1)
minorx = np.arange(-6, 6, 0.5)
majory = np.arange(-6, 7, 1)
minory = np.arange(-6, 6, 0.5)
self.widget_4.canvas.axes.set_xticks(majorx)
self.widget_4.canvas.axes.set_xticks(minorx, minor = True)
self.widget_4.canvas.axes.set_yticks(minory, minor = True)
self.widget_4.canvas.axes.set_yticks(majory)
self.widget_4.canvas.axes.grid(which='both')
self.widget_4.canvas.axes.grid(which='minor')
self.widget_4.canvas.axes.grid(which='major')
self.widget_4.canvas.axes.grid(True)
#Mostrar gráfica
self.widget_4.canvas.draw()
#Actualización de la desviación adquirida
self.lineEdit_17.setText(str(round(xs, 3)))
self.lineEdit_18.setText(str(round(ys, 3)))
#Actualización desviaciones promedio
self.lineEdit_25.setText(str(round(self.VarAz/self.number, 2)))
self.lineEdit_26.setText(str(round(self.VarEl/self.number, 2)))
self.pushButton_8.setEnabled(True)
cb.remove()
#Cierre ventana espera
Wait.close()

```

else:

```

#Mensaje advertencia para coordenadas erroneas
QMessageBox.about(self , "Advertencia" , \
                  "Las coordenadas ingresadas están fuera del rango" \
                  "o son de número decimal impar")

```

#Cierre de la aplicación

```

def main():
    app = QtWidgets.QApplication(sys.argv)
    main = MainWindow()
    main.show()
    sys.exit(app.exec_())
#Ejecución de la aplicación
if __name__ == "__main__":
    app = QtWidgets.QApplication([])
    app.setWindowIcon(QtGui.QIcon('/:icons/Logoapp.ico'))
    window = MainWindow()
    window.show()
    app.exec_()

```

B.3 Código fuente entorno gráfico “Adquisicion.py”

```

from PyQt5 import QtCore, QtGui, QtWidgets

class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        MainWindow.setObjectName("MainWindow")
        MainWindow.resize(947, 643)
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.tabWidget = QtWidgets.QTabWidget(self.centralwidget)
        self.tabWidget.setGeometry(QtCore.QRect(0, 10, 941, 601))
        self.tabWidget.setObjectName("tabWidget")
        self.tab = QtWidgets.QWidget()
        self.tab.setObjectName("tab")
        self.widget = PlotWidget(self.tab)
        self.widget.setGeometry(QtCore.QRect(10, 20, 281, 231))
        self.widget.setObjectName("widget")
        self.widget_2 = PlotWidget(self.tab)
        self.widget_2.setGeometry(QtCore.QRect(310, 20, 281, 231))
        self.widget_2.setObjectName("widget_2")
        self.widget_3 = PlotWidget(self.tab)
        self.widget_3.setGeometry(QtCore.QRect(610, 20, 281, 231))
        self.widget_3.setObjectName("widget_3")

```

```

self.pushButton = QtWidgets.QPushButton(self.tab)
self.pushButton.setGeometry(QtCore.QRect(114, 260, 91, 23))
self.pushButton.setObjectName("pushButton")
self.pushButton_2 = QtWidgets.QPushButton(self.tab)
self.pushButton_2.setGeometry(QtCore.QRect(420, 260, 81, 23))
self.pushButton_2.setObjectName("pushButton_2")
self.pushButton_3 = QtWidgets.QPushButton(self.tab)
self.pushButton_3.setGeometry(QtCore.QRect(720, 260, 81, 23))
self.pushButton_3.setObjectName("pushButton_3")
self.groupBox = QtWidgets.QGroupBox(self.tab)
self.groupBox.setGeometry(QtCore.QRect(110, 320, 311, 151))
self.groupBox.setObjectName("groupBox")
self.pushButton_4 = QtWidgets.QPushButton(self.groupBox)
self.pushButton_4.setGeometry(QtCore.QRect(20, 120, 75, 23))
self.pushButton_4.setObjectName("pushButton_4")
self.lineEdit = QtWidgets.QLineEdit(self.groupBox)
self.lineEdit.setGeometry(QtCore.QRect(20, 90, 71, 20))
self.lineEdit.setObjectName("lineEdit")
self.lineEdit_2 = QtWidgets.QLineEdit(self.groupBox)
self.lineEdit_2.setGeometry(QtCore.QRect(20, 40, 71, 20))
self.lineEdit_2.setObjectName("lineEdit_2")
self.label = QtWidgets.QLabel(self.groupBox)
self.label.setGeometry(QtCore.QRect(20, 20, 47, 13))
self.label.setObjectName("label")
self.label_2 = QtWidgets.QLabel(self.groupBox)
self.label_2.setGeometry(QtCore.QRect(20, 70, 47, 13))
self.label_2.setObjectName("label_2")
self.label_3 = QtWidgets.QLabel(self.groupBox)
self.label_3.setGeometry(QtCore.QRect(200, 20, 81, 16))
self.label_3.setObjectName("label_3")
self.label_5 = QtWidgets.QLabel(self.groupBox)
self.label_5.setGeometry(QtCore.QRect(200, 90, 47, 13))
self.label_5.setObjectName("label_5")
self.lineEdit_3 = QtWidgets.QLineEdit(self.groupBox)
self.lineEdit_3.setEnabled(False)
self.lineEdit_3.setGeometry(QtCore.QRect(200, 110, 71, 20))
self.lineEdit_3.setObjectName("lineEdit_3")

```

```

self.lineEdit_4 = QtWidgets.QLineEdit( self.groupBox)
self.lineEdit_4.setEnabled( False)
self.lineEdit_4.setGeometry( QtCore.QRect(200, 60, 71, 20))
self.lineEdit_4.setObjectName("lineEdit_4")
self.label_4 = QtWidgets.QLabel( self.groupBox)
self.label_4.setGeometry( QtCore.QRect(200, 40, 47, 13))
self.label_4.setObjectName("label_4")
self.label_6 = QtWidgets.QLabel( self.tab)
self.label_6.setGeometry( QtCore.QRect(470, 340, 47, 13))
self.label_6.setObjectName("label_6")
self.lineEdit_5 = QtWidgets.QLineEdit( self.tab)
self.lineEdit_5.setEnabled( False)
self.lineEdit_5.setGeometry( QtCore.QRect(440, 360, 81, 20))
self.lineEdit_5.setObjectName("lineEdit_5")
self.lineEdit_27 = QtWidgets.QLineEdit( self.tab)
self.lineEdit_27.setEnabled( False)
self.lineEdit_27.setGeometry( QtCore.QRect(120, 290, 81, 20))
self.lineEdit_27.setObjectName("lineEdit_27")
self.lineEdit_28 = QtWidgets.QLineEdit( self.tab)
self.lineEdit_28.setEnabled( False)
self.lineEdit_28.setGeometry( QtCore.QRect(420, 290, 81, 20))
self.lineEdit_28.setObjectName("lineEdit_28")
self.lineEdit_29 = QtWidgets.QLineEdit( self.tab)
self.lineEdit_29.setEnabled( False)
self.lineEdit_29.setGeometry( QtCore.QRect(720, 290, 81, 20))
self.lineEdit_29.setObjectName("lineEdit_29")
self.lineEdit_22 = QtWidgets.QLineEdit( self.tab)
self.lineEdit_22.setEnabled( False)
self.lineEdit_22.setGeometry( QtCore.QRect(440, 390, 81, 20))
self.lineEdit_22.setObjectName("lineEdit_22")
self.label_12 = QtWidgets.QLabel( self.tab)
self.label_12.setGeometry( QtCore.QRect(110, 470, 121, 16))
self.label_12.setObjectName("label_12")
self.label_13 = QtWidgets.QLabel( self.tab)
self.label_13.setGeometry( QtCore.QRect(110, 490, 121, 16))
self.label_13.setObjectName("label_13")
self.lineEdit_23 = QtWidgets.QLineEdit( self.tab)

```



```

self.lineEdit_23.setGeometry(QtCore.QRect(250, 470, 71, 20))
self.lineEdit_23.setObjectName("lineEdit_23")
self.lineEdit_24 = QtWidgets.QLineEdit(self.tab)
self.lineEdit_24.setGeometry(QtCore.QRect(250, 490, 71, 20))
self.lineEdit_24.setObjectName("lineEdit_24")
self.label_14 = QtWidgets.QLabel(self.tab)
self.label_14.setGeometry(QtCore.QRect(540, 335, 141, 21))
self.label_14.setObjectName("label_14")
self.label_15 = QtWidgets.QLabel(self.tab)
self.label_15.setGeometry(QtCore.QRect(700, 330, 141, 31))
self.label_15.setObjectName("label_15")
self.lineEdit_10 = QtWidgets.QLineEdit(self.tab)
self.lineEdit_10.setEnabled(False)
self.lineEdit_10.setGeometry(QtCore.QRect(560, 360, 81, 20))
self.lineEdit_10.setObjectName("lineEdit_10")
self.lineEdit_21 = QtWidgets.QLineEdit(self.tab)
self.lineEdit_21.setEnabled(False)
self.lineEdit_21.setGeometry(QtCore.QRect(730, 360, 81, 20))
self.lineEdit_21.setObjectName("lineEdit_21")
self.label_16 = QtWidgets.QLabel(self.tab)
self.label_16.setGeometry(QtCore.QRect(440, 430, 91, 16))
self.label_16.setObjectName("label_16")
self.pushButton_7 = QtWidgets.QPushButton(self.tab)
self.pushButton_7.setGeometry(QtCore.QRect(440, 460, 75, 23))
self.pushButton_7.setObjectName("pushButton_7")
self.label_46 = QtWidgets.QLabel(self.tab)
self.label_46.setGeometry(QtCore.QRect(110, 510, 131, 16))
self.label_46.setObjectName("label_46")
self.lineEdit_30 = QtWidgets.QLineEdit(self.tab)
self.lineEdit_30.setGeometry(QtCore.QRect(250, 510, 71, 20))
self.lineEdit_30.setObjectName("lineEdit_30")
self.lineEdit_31 = QtWidgets.QLineEdit(self.tab)
self.lineEdit_31.setGeometry(QtCore.QRect(250, 530, 71, 20))
self.lineEdit_31.setObjectName("lineEdit_31")
self.label_47 = QtWidgets.QLabel(self.tab)
self.label_47.setGeometry(QtCore.QRect(110, 530, 141, 16))
self.label_47.setObjectName("label_47")

```

```

self.lineEdit_32 = QtWidgets.QLineEdit(self.tab)
self.lineEdit_32.setEnabled(False)
self.lineEdit_32.setGeometry(QtCore.QRect(650, 425, 81, 20))
self.lineEdit_32.setObjectName("lineEdit_32")
self.label_48 = QtWidgets.QLabel(self.tab)
self.label_48.setGeometry(QtCore.QRect(630, 400, 141, 21))
self.label_48.setObjectName("label_48")
self.tabWidget.addTab(self.tab, "")
self.tab_2 = QtWidgets.QWidget()
self.tab_2.setObjectName("tab_2")
self.groupBox_2 = QtWidgets.QGroupBox(self.tab_2)
self.groupBox_2.setGeometry(QtCore.QRect(10, 10, 271, 151))
self.groupBox_2.setObjectName("groupBox_2")
self.pushButton_5 = QtWidgets.QPushButton(self.groupBox_2)
self.pushButton_5.setGeometry(QtCore.QRect(20, 120, 75, 23))
self.pushButton_5.setObjectName("pushButton_5")
self.lineEdit_6 = QtWidgets.QLineEdit(self.groupBox_2)
self.lineEdit_6.setGeometry(QtCore.QRect(20, 90, 71, 20))
self.lineEdit_6.setObjectName("lineEdit_6")
self.lineEdit_7 = QtWidgets.QLineEdit(self.groupBox_2)
self.lineEdit_7.setGeometry(QtCore.QRect(20, 40, 71, 20))
self.lineEdit_7.setObjectName("lineEdit_7")
self.label_7 = QtWidgets.QLabel(self.groupBox_2)
self.label_7.setGeometry(QtCore.QRect(20, 20, 47, 13))
self.label_7.setObjectName("label_7")
self.label_8 = QtWidgets.QLabel(self.groupBox_2)
self.label_8.setGeometry(QtCore.QRect(20, 70, 47, 13))
self.label_8.setObjectName("label_8")
self.label_9 = QtWidgets.QLabel(self.groupBox_2)
self.label_9.setGeometry(QtCore.QRect(140, 20, 81, 16))
self.label_9.setObjectName("label_9")
self.label_10 = QtWidgets.QLabel(self.groupBox_2)
self.label_10.setGeometry(QtCore.QRect(140, 90, 47, 13))
self.label_10.setObjectName("label_10")
self.lineEdit_8 = QtWidgets.QLineEdit(self.groupBox_2)
self.lineEdit_8.setEnabled(False)
self.lineEdit_8.setGeometry(QtCore.QRect(140, 110, 71, 20))

```

```

self.lineEdit_8.setObjectName("lineEdit_8")
self.lineEdit_9 = QtWidgets.QLineEdit(self.groupBox_2)
self.lineEdit_9.setEnabled(False)
self.lineEdit_9.setGeometry(QtCore.QRect(140, 60, 71, 20))
self.lineEdit_9.setObjectName("lineEdit_9")
self.label_11 = QtWidgets.QLabel(self.groupBox_2)
self.label_11.setGeometry(QtCore.QRect(140, 40, 47, 13))
self.label_11.setObjectName("label_11")
self.label_17 = QtWidgets.QLabel(self.groupBox_2)
self.label_17.setGeometry(QtCore.QRect(10, 160, 47, 13))
self.label_17.setObjectName("label_17")
self.widget_6 = MplWidget(self.tab_2)
self.widget_6.setGeometry(QtCore.QRect(360, 0, 571, 541))
self.widget_6.setObjectName("widget_6")
self.Grafica = QtWidgets.QLabel(self.tab_2)
self.Grafica.setGeometry(QtCore.QRect(30, 270, 301, 241))
self.Grafica.setText("")
self.Grafica.setObjectName("Grafica")
self.label_18 = QtWidgets.QLabel(self.tab_2)
self.label_18.setGeometry(QtCore.QRect(30, 170, 47, 21))
self.label_18.setObjectName("label_18")
self.Picsize = QtWidgets.QLineEdit(self.tab_2)
self.Picsize.setGeometry(QtCore.QRect(80, 170, 61, 20))
self.Picsize.setObjectName("Picsize")
self.label_19 = QtWidgets.QLabel(self.tab_2)
self.label_19.setGeometry(QtCore.QRect(150, 170, 61, 16))
self.label_19.setObjectName("label_19")
self.label_20 = QtWidgets.QLabel(self.tab_2)
self.label_20.setGeometry(QtCore.QRect(150, 190, 61, 16))
self.label_20.setObjectName("label_20")
self.lineEdit_20 = QtWidgets.QLineEdit(self.tab_2)
self.lineEdit_20.setEnabled(False)
self.lineEdit_20.setGeometry(QtCore.QRect(210, 170, 71, 20))
self.lineEdit_20.setObjectName("lineEdit_20")
self.lineEdit_19 = QtWidgets.QLineEdit(self.tab_2)
self.lineEdit_19.setEnabled(False)
self.lineEdit_19.setGeometry(QtCore.QRect(210, 190, 71, 20))

```

```

self.lineEdit_19.setObjectName("lineEdit_19")
self.label_21 = QtWidgets.QLabel(self.tab_2)
self.label_21.setGeometry(QtCore.QRect(30, 220, 101, 21))
self.label_21.setObjectName("label_21")
self.comboBox_3 = QtWidgets.QComboBox(self.tab_2)
self.comboBox_3.setGeometry(QtCore.QRect(150, 220, 61, 22))
self.comboBox_3.setObjectName("comboBox_3")
self.comboBox_3.addItem("")
self.comboBox_3.addItem("")
self.comboBox_3.addItem("")
self.Picsize_2 = QtWidgets.QLineEdit(self.tab_2)
self.Picsize_2.setGeometry(QtCore.QRect(80, 190, 61, 20))
self.Picsize_2.setObjectName("Picsize_2")
self.label_49 = QtWidgets.QLabel(self.tab_2)
self.label_49.setGeometry(QtCore.QRect(30, 190, 47, 21))
self.label_49.setObjectName("label_49")
self.label_50 = QtWidgets.QLabel(self.tab_2)
self.label_50.setGeometry(QtCore.QRect(30, 200, 47, 21))
self.label_50.setObjectName("label_50")
self.tabWidget.addTab(self.tab_2, "")
self.tab_3 = QtWidgets.QWidget()
self.tab_3.setObjectName("tab_3")
self.groupBox_3 = QtWidgets.QGroupBox(self.tab_3)
self.groupBox_3.setGeometry(QtCore.QRect(30, 10, 341, 161))
self.groupBox_3.setObjectName("groupBox_3")
self.pushButton_6 = QtWidgets.QPushButton(self.groupBox_3)
self.pushButton_6.setGeometry(QtCore.QRect(20, 120, 75, 23))
self.pushButton_6.setObjectName("pushButton_6")
self.lineEdit_11 = QtWidgets.QLineEdit(self.groupBox_3)
self.lineEdit_11.setGeometry(QtCore.QRect(20, 90, 71, 20))
self.lineEdit_11.setObjectName("lineEdit_11")
self.lineEdit_12 = QtWidgets.QLineEdit(self.groupBox_3)
self.lineEdit_12.setGeometry(QtCore.QRect(20, 40, 71, 20))
self.lineEdit_12.setObjectName("lineEdit_12")
self.label_31 = QtWidgets.QLabel(self.groupBox_3)
self.label_31.setGeometry(QtCore.QRect(20, 20, 47, 13))
self.label_31.setObjectName("label_31")

```

```

self.label_32 = QtWidgets.QLabel(self.groupBox_3)
self.label_32.setGeometry(QtCore.QRect(20, 70, 47, 13))
self.label_32.setObjectName("label_32")
self.label_33 = QtWidgets.QLabel(self.groupBox_3)
self.label_33.setGeometry(QtCore.QRect(140, 10, 81, 16))
self.label_33.setObjectName("label_33")
self.label_34 = QtWidgets.QLabel(self.groupBox_3)
self.label_34.setGeometry(QtCore.QRect(140, 90, 47, 13))
self.label_34.setObjectName("label_34")
self.lineEdit_13 = QtWidgets.QLineEdit(self.groupBox_3)
self.lineEdit_13.setEnabled(False)
self.lineEdit_13.setGeometry(QtCore.QRect(140, 110, 71, 20))
self.lineEdit_13.setObjectName("lineEdit_13")
self.lineEdit_14 = QtWidgets.QLineEdit(self.groupBox_3)
self.lineEdit_14.setEnabled(False)
self.lineEdit_14.setGeometry(QtCore.QRect(140, 60, 71, 20))
self.lineEdit_14.setObjectName("lineEdit_14")
self.label_35 = QtWidgets.QLabel(self.groupBox_3)
self.label_35.setGeometry(QtCore.QRect(140, 40, 47, 13))
self.label_35.setObjectName("label_35")
self.label_36 = QtWidgets.QLabel(self.groupBox_3)
self.label_36.setGeometry(QtCore.QRect(10, 160, 47, 13))
self.label_36.setObjectName("label_36")
self.label_37 = QtWidgets.QLabel(self.groupBox_3)
self.label_37.setGeometry(QtCore.QRect(240, 10, 41, 16))
self.label_37.setObjectName("label_37")
self.label_38 = QtWidgets.QLabel(self.groupBox_3)
self.label_38.setGeometry(QtCore.QRect(240, 35, 91, 21))
self.label_38.setObjectName("label_38")
self.label_39 = QtWidgets.QLabel(self.groupBox_3)
self.label_39.setGeometry(QtCore.QRect(240, 80, 51, 31))
self.label_39.setObjectName("label_39")
self.comboBox = QtWidgets.QComboBox(self.groupBox_3)
self.comboBox.setGeometry(QtCore.QRect(240, 60, 69, 22))
self.comboBox.setObjectName("comboBox")
self.comboBox.addItem("")
self.comboBox.addItem("")

```

```

self.comboBox.addItem("")
self.comboBox.addItem("")
self.comboBox_2 = QtWidgets.QComboBox(self.groupBox_3)
self.comboBox_2.setGeometry(QtCore.QRect(240, 110, 69, 22))
self.comboBox_2.setObjectName("comboBox_2")
self.comboBox_2.addItem("")
self.comboBox_2.addItem("")
self.comboBox_2.addItem("")
self.label_40 = QtWidgets.QLabel(self.tab_3)
self.label_40.setGeometry(QtCore.QRect(30, 170, 61, 16))
self.label_40.setObjectName("label_40")
self.label_41 = QtWidgets.QLabel(self.tab_3)
self.label_41.setGeometry(QtCore.QRect(200, 170, 61, 16))
self.label_41.setObjectName("label_41")
self.lineEdit_15 = QtWidgets.QLineEdit(self.tab_3)
self.lineEdit_15.setEnabled(False)
self.lineEdit_15.setGeometry(QtCore.QRect(110, 170, 81, 20))
self.lineEdit_15.setObjectName("lineEdit_15")
self.lineEdit_16 = QtWidgets.QLineEdit(self.tab_3)
self.lineEdit_16.setEnabled(False)
self.lineEdit_16.setGeometry(QtCore.QRect(290, 170, 81, 20))
self.lineEdit_16.setObjectName("lineEdit_16")
self.label_42 = QtWidgets.QLabel(self.tab_3)
self.label_42.setGeometry(QtCore.QRect(30, 200, 91, 16))
self.label_42.setObjectName("label_42")
self.lineEdit_17 = QtWidgets.QLineEdit(self.tab_3)
self.lineEdit_17.setEnabled(False)
self.lineEdit_17.setGeometry(QtCore.QRect(130, 200, 61, 20))
self.lineEdit_17.setText("")
self.lineEdit_17.setObjectName("lineEdit_17")
self.label_43 = QtWidgets.QLabel(self.tab_3)
self.label_43.setGeometry(QtCore.QRect(200, 200, 101, 16))
self.label_43.setObjectName("label_43")
self.lineEdit_18 = QtWidgets.QLineEdit(self.tab_3)
self.lineEdit_18.setEnabled(False)
self.lineEdit_18.setGeometry(QtCore.QRect(310, 200, 61, 20))
self.lineEdit_18.setText("")

```

```

self.lineEdit_18.setObjectName("lineEdit_18")
self.lineEdit_26 = QtWidgets.QLineEdit(self.tab_3)
self.lineEdit_26.setEnabled(False)
self.lineEdit_26.setGeometry(QtCore.QRect(330, 230, 41, 20))
self.lineEdit_26.setText("")
self.lineEdit_26.setObjectName("lineEdit_26")
self.label_44 = QtWidgets.QLabel(self.tab_3)
self.label_44.setGeometry(QtCore.QRect(30, 230, 121, 16))
self.label_44.setObjectName("label_44")
self.lineEdit_25 = QtWidgets.QLineEdit(self.tab_3)
self.lineEdit_25.setEnabled(False)
self.lineEdit_25.setGeometry(QtCore.QRect(150, 230, 41, 20))
self.lineEdit_25.setText("")
self.lineEdit_25.setObjectName("lineEdit_25")
self.label_45 = QtWidgets.QLabel(self.tab_3)
self.label_45.setGeometry(QtCore.QRect(200, 230, 131, 16))
self.label_45.setObjectName("label_45")
self.widget_5 = MplWidget(self.tab_3)
self.widget_5.setGeometry(QtCore.QRect(390, 0, 551, 561))
self.widget_5.setObjectName("widget_5")
self.widget_4 = MplWidget(self.tab_3)
self.widget_4.setGeometry(QtCore.QRect(0, 280, 391, 281))
self.widget_4.setObjectName("widget_4")
self.pushButton_8 = QtWidgets.QPushButton(self.tab_3)
self.pushButton_8.setEnabled(False)
self.pushButton_8.setGeometry(QtCore.QRect(180, 250, 50, 23))
self.pushButton_8.setObjectName("pushButton_8")
self.tabWidget.addTab(self.tab_3, "")
MainWindow.setCentralWidget(self.centralwidget)
self.menubar = QtWidgets.QMenuBar(MainWindow)
self.menubar.setGeometry(QtCore.QRect(0, 0, 947, 21))
self.menubar.setObjectName("menubar")
self.menuAyuda = QtWidgets.QMenu(self.menubar)
self.menuAyuda.setObjectName("menuAyuda")
MainWindow.setMenuBar(self.menubar)
self.statusbar = QtWidgets.QStatusBar(MainWindow)
self.statusbar.setObjectName("statusbar")

```

```

MainWindow.setStatusBar(self.statusbar)
self.AcercaDe = QtWidgets.QAction(MainWindow)
self.AcercaDe.setObjectName("AcercaDe")
self.GuiaLink = QtWidgets.QAction(MainWindow)
self.GuiaLink.setObjectName("GuiaLink")
self.Guia_0 = QtWidgets.QAction(MainWindow)
self.Guia_0.setObjectName("Guia_0")
self.menuAyuda.addAction(self.AcercaDe)
self.menuAyuda.addAction(self.Guia_0)
self.menubar.addAction(self.menuAyuda.menuAction())

self.retranslateUi(MainWindow)
self.tabWidget.setCurrentIndex(0)
QtCore.QMetaObject.connectSlotsByName(MainWindow)

def retranslateUi(self, MainWindow):
    _translate = QtCore.QCoreApplication.translate
    MainWindow.setWindowTitle(_translate("MainWindow", "CRT"))
    self.pushButton.setText(_translate("MainWindow", "Cielo_de_fondo"))
    self.pushButton_2.setText(_translate("MainWindow", "Referencia"))
    self.pushButton_3.setText(_translate("MainWindow", "Tránsito"))
    self.groupBox.setTitle(_translate("MainWindow", "Posicionamiento"))
    self.pushButton_4.setText(_translate("MainWindow", "Enviar"))
    self.lineEdit.setText(_translate("MainWindow", "90.0"))
    self.lineEdit_2.setText(_translate("MainWindow", "0.0"))
    self.label.setText(_translate("MainWindow", "Azimut"))
    self.label_2.setText(_translate("MainWindow", "Elevación"))
    self.label_3.setText(_translate("MainWindow", "Posición_actual"))
    self.label_5.setText(_translate("MainWindow", "Elevación"))
    self.lineEdit_3.setText(_translate("MainWindow", "90.0"))
    self.lineEdit_4.setText(_translate("MainWindow", "0.0"))
    self.label_4.setText(_translate("MainWindow", "Azimut"))
    self.label_6.setText(_translate("MainWindow", "HPBW"))
    self.label_12.setText(_translate("MainWindow", \
        "Temperatura_Cúpula_(K)"))
    self.label_13.setText(_translate("MainWindow", \
        "Declinación_Objeto_(°)"))

```



```

self.lineEdit_23.setText(_translate("MainWindow", "250.0"))
self.lineEdit_24.setText(_translate("MainWindow", "0.0"))
self.label_14.setText(_translate("MainWindow", \
    "Temperatura_del_sistema_(K)"))
self.label_15.setText(_translate("MainWindow", \
    "Temperatura_observada_(K)"))
self.label_16.setText(_translate("MainWindow", \
    "Ajuste_Gaussiano"))
self.pushButton_7.setText(_translate("MainWindow", "Ver"))
self.label_46.setText(_translate("MainWindow", \
    "Tiempo_de_Adquisición_(Min)"))
self.lineEdit_30.setText(_translate("MainWindow", "15"))
self.lineEdit_31.setText(_translate("MainWindow", "0.5"))
self.label_47.setText(_translate("MainWindow", \
    "Ancho_angular_del_objeto_(°)"))
self.label_48.setText(_translate("MainWindow", \
    "Temperatura_de_brillo_(K)"))
self.tabWidget.setTabText(self.tabWidget.indexOf(self.tab),\
    _translate("MainWindow", "Adquisicion_de_datos"))
self.groupBox_2.setTitle(_translate("MainWindow", \
    "Posicionamiento"))
self.pushButton_5.setText(_translate("MainWindow", "Enviar"))
self.lineEdit_6.setText(_translate("MainWindow", "90.0"))
self.lineEdit_7.setText(_translate("MainWindow", "0.0"))
self.label_7.setText(_translate("MainWindow", "Azimut"))
self.label_8.setText(_translate("MainWindow", \
    "Elevación"))
self.label_9.setText(_translate("MainWindow", \
    "Posición_actual"))
self.label_10.setText(_translate("MainWindow", "Elevación"))
self.lineEdit_8.setText(_translate("MainWindow", "90.0"))
self.lineEdit_9.setText(_translate("MainWindow", "0.0"))
self.label_11.setText(_translate("MainWindow", "Azimut"))
self.label_17.setText(_translate("MainWindow", "Azimut"))
self.label_18.setText(_translate("MainWindow", "Pic-Size:"))
self.Picsize.setText(_translate("MainWindow", "50"))
self.label_19.setText(_translate("MainWindow", "Hora_Inicio:"))

```

```

self.label_20.setText(_translate("MainWindow", "Hora_Final:"))
self.label_21.setText(_translate("MainWindow", \
    "Tamaño_de_ejes_(°):"))
self.comboBox_3.setItemText(0, _translate("MainWindow", "2"))
self.comboBox_3.setItemText(1, _translate("MainWindow", "4"))
self.comboBox_3.setItemText(2, _translate("MainWindow", "6"))
self.Picsize_2.setText(_translate("MainWindow", "10"))
self.label_49.setText(_translate("MainWindow", "Sample:"))
self.label_50.setText(_translate("MainWindow", "rate"))
self.tabWidget.setTabText(self.tabWidget.indexOf(self.tab_2), \
    _translate("MainWindow", "Escaneo"))
self.groupBox_3.setTitle(_translate("MainWindow", \
    "Posicionamiento"))
self.pushButton_6.setText(_translate("MainWindow", "Enviar"))
self.lineEdit_11.setText(_translate("MainWindow", "90.0"))
self.lineEdit_12.setText(_translate("MainWindow", "0.0"))
self.label_31.setText(_translate("MainWindow", "Azimut"))
self.label_32.setText(_translate("MainWindow", "Elevación"))
self.label_33.setText(_translate("MainWindow", \
    "Posición_actual"))
self.label_34.setText(_translate("MainWindow", "Elevación"))
self.lineEdit_13.setText(_translate("MainWindow", "90.0"))
self.lineEdit_14.setText(_translate("MainWindow", "0.0"))
self.label_35.setText(_translate("MainWindow", "Azimut"))
self.label_36.setText(_translate("MainWindow", "Azimut"))
self.label_37.setText(_translate("MainWindow", "Ventana"))
self.label_38.setText(_translate("MainWindow", \
    "Tamajo_de_ejes_(°)"))
self.label_39.setText(_translate("MainWindow", "Paso_(°)"))
self.comboBox.setItemText(0, _translate("MainWindow", "2"))
self.comboBox.setItemText(1, _translate("MainWindow", "4"))
self.comboBox.setItemText(2, _translate("MainWindow", "8"))
self.comboBox.setItemText(3, _translate("MainWindow", "12"))
self.comboBox_2.setItemText(0, _translate("MainWindow", "0.2"))
self.comboBox_2.setItemText(1, _translate("MainWindow", "0.4"))
self.comboBox_2.setItemText(2, _translate("MainWindow", "1"))
self.label_40.setText(_translate("MainWindow", "Hora_Inicio:"))

```

```

self.label_41.setText(_translate("MainWindow", "Hora_Final:"))
self.label_42.setText(_translate("MainWindow", \
    "Desv. Azimut(°):"))
self.label_43.setText(_translate("MainWindow", \
    "Desv. Elevación(°):"))
self.label_44.setText(_translate("MainWindow", \
    "Desv. Azimut_Prom.(°):"))
self.label_45.setText(_translate("MainWindow", \
    "Desv. Elevación_Prom.(°):"))
self.pushButton_8.setText(_translate("MainWindow", \
    "Reiniciar"))
self.tabWidget.setTabText(self.tabWidget.indexOf(self.tab_3), \
    _translate("MainWindow", "Calibración"))
self.menuAyuda.setTitle(_translate("MainWindow", "Ayuda"))
self.AcercaDe.setText(_translate("MainWindow", "Acerca_de"))
self.GuiaLink.setText(_translate("MainWindow", "Link"))
self.Guia_0.setText(_translate("MainWindow", "Guia"))
from mplwidget import MplWidget
from pyqtgraph import PlotWidget

```

B.4 Código fuente widget-matplotlib “mplwidget.py”

```
from PyQt5.QtWidgets import*

from matplotlib.backends.backend_qt5agg import (FigureCanvas,
        NavigationToolbar2QT as NavigationToolbar)

from matplotlib.figure import Figure

class MplWidget(QWidget):
    def __init__(self, parent = None):
        QWidget.__init__(self, parent)
        self.fig = Figure()
        self.fig.clf()
        self.canvas = FigureCanvas(self.fig)
        self.toolbar = NavigationToolbar(self.canvas, self)
        vertical_layout = QVBoxLayout()
        vertical_layout.addWidget(self.canvas)
        vertical_layout.addWidget(self.toolbar)
        self.fig.subplots_adjust(left=0.150, right=0.940,
            top=0.980, bottom=0.185)
        self.canvas.axes = self.canvas.figure.add_subplot(111)
        self.setLayout(vertical_layout)
```